PICAXE Interfacing and Programming

By Darrell Davis KT4WX
ARRL Technical Specialist
ARRL West Central Florida Section

PICAXE Interfacing and Programming

KEY TO THIS PROGRAM

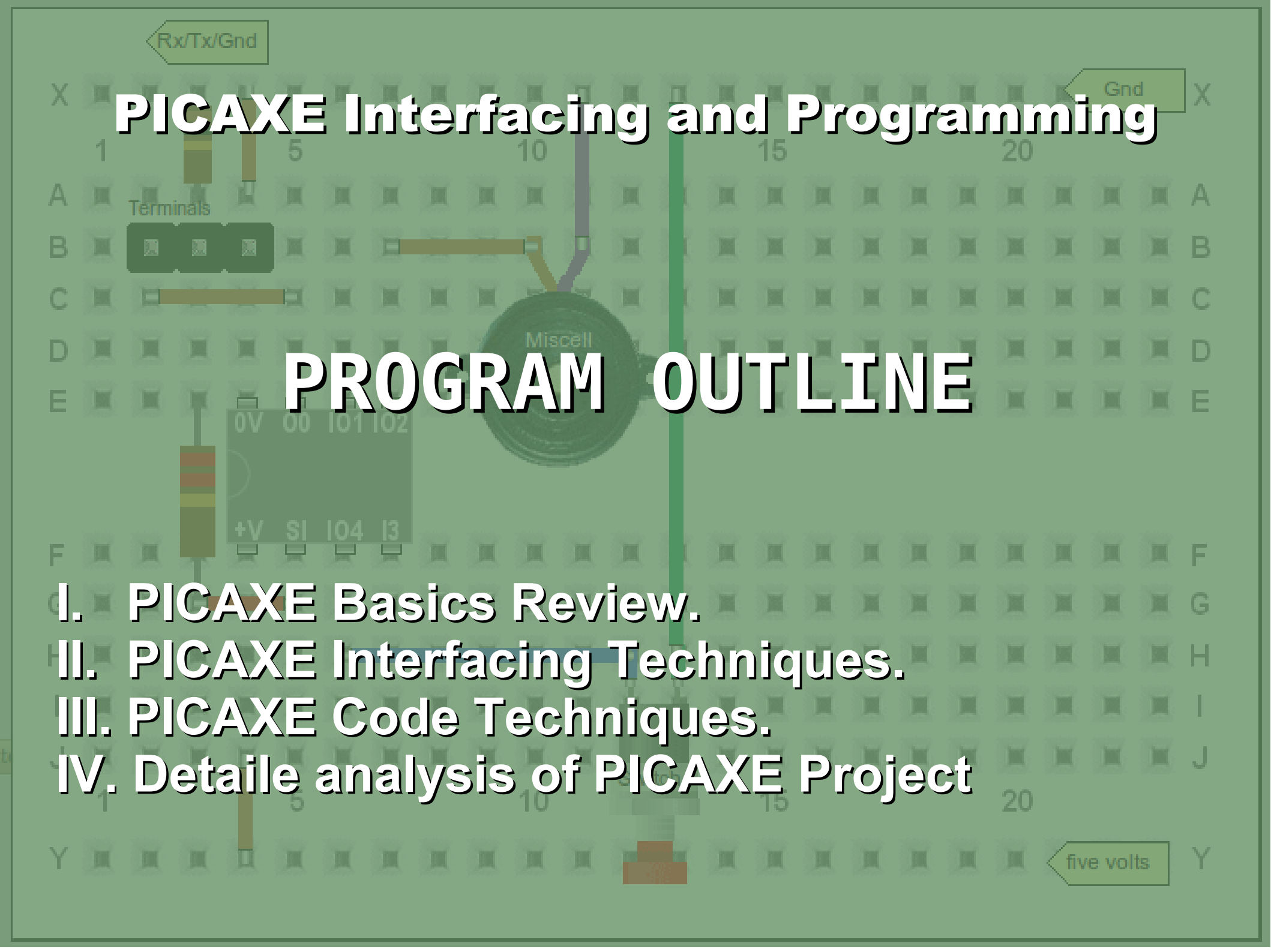
If you like this program:
Fall asleep with your head forward

If you dislike this program:
Fall asleep with your head backward.

All the heads should be forward
when I am finished :-)

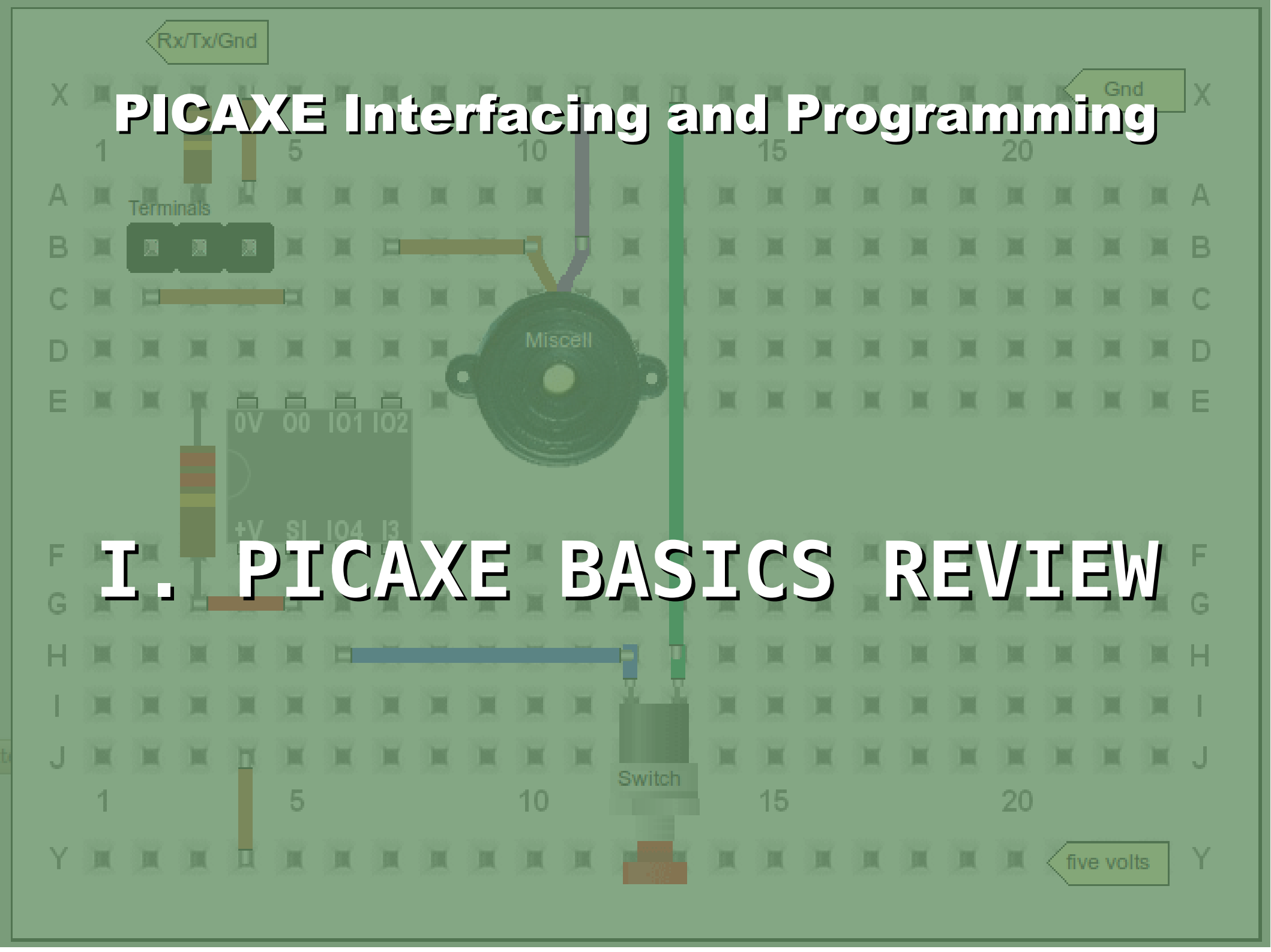
- Ed Hare W1RFI, 2001

PICAXE Interfacing and Programming



PROGRAM OUTLINE

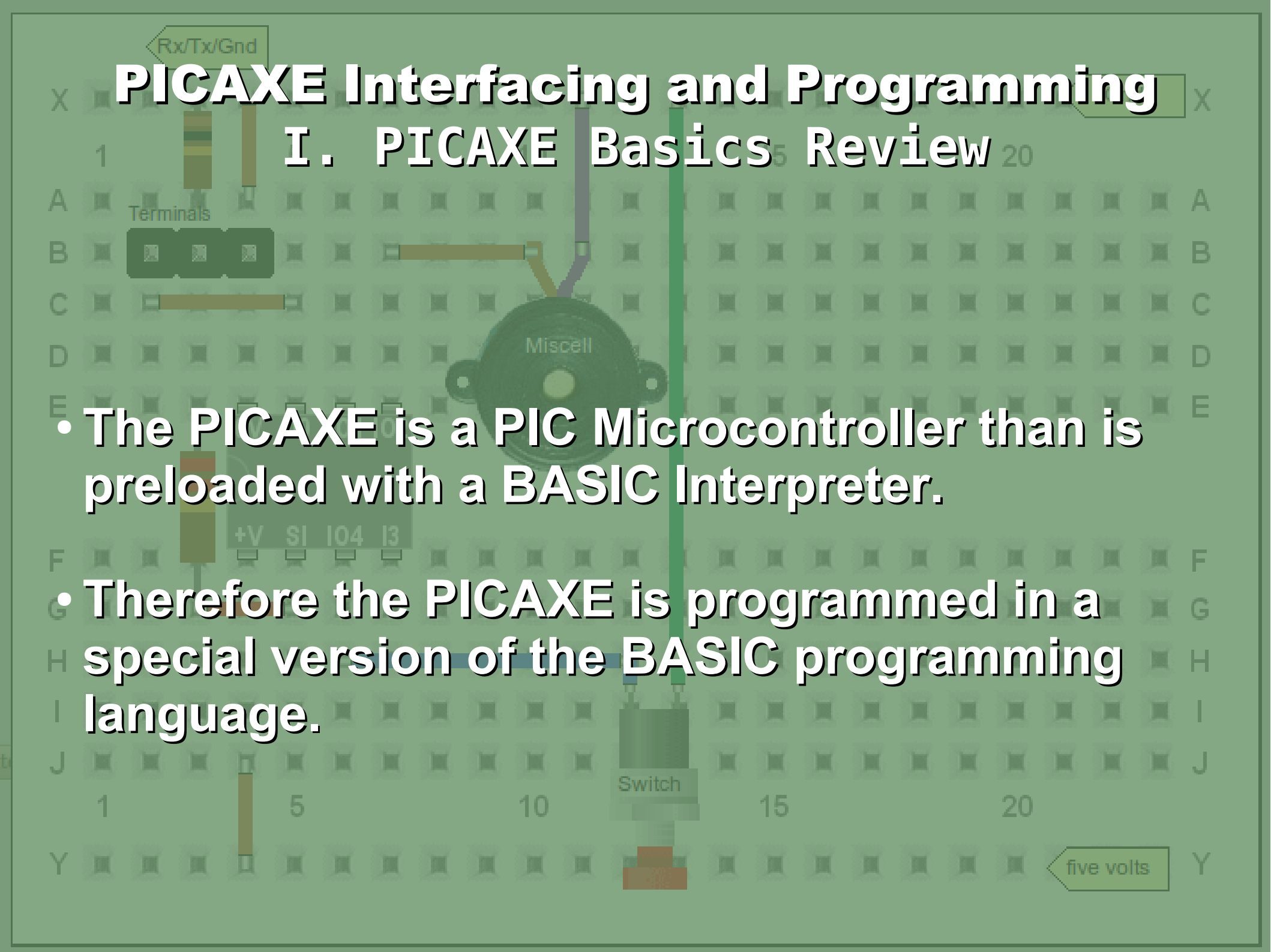
I. PICAXE Basics Review.
II. PICAXE Interfacing Techniques.
III. PICAXE Code Techniques.
IV. Detaile analysis of PICAXE Project



PICAXE Interfacing and Programming

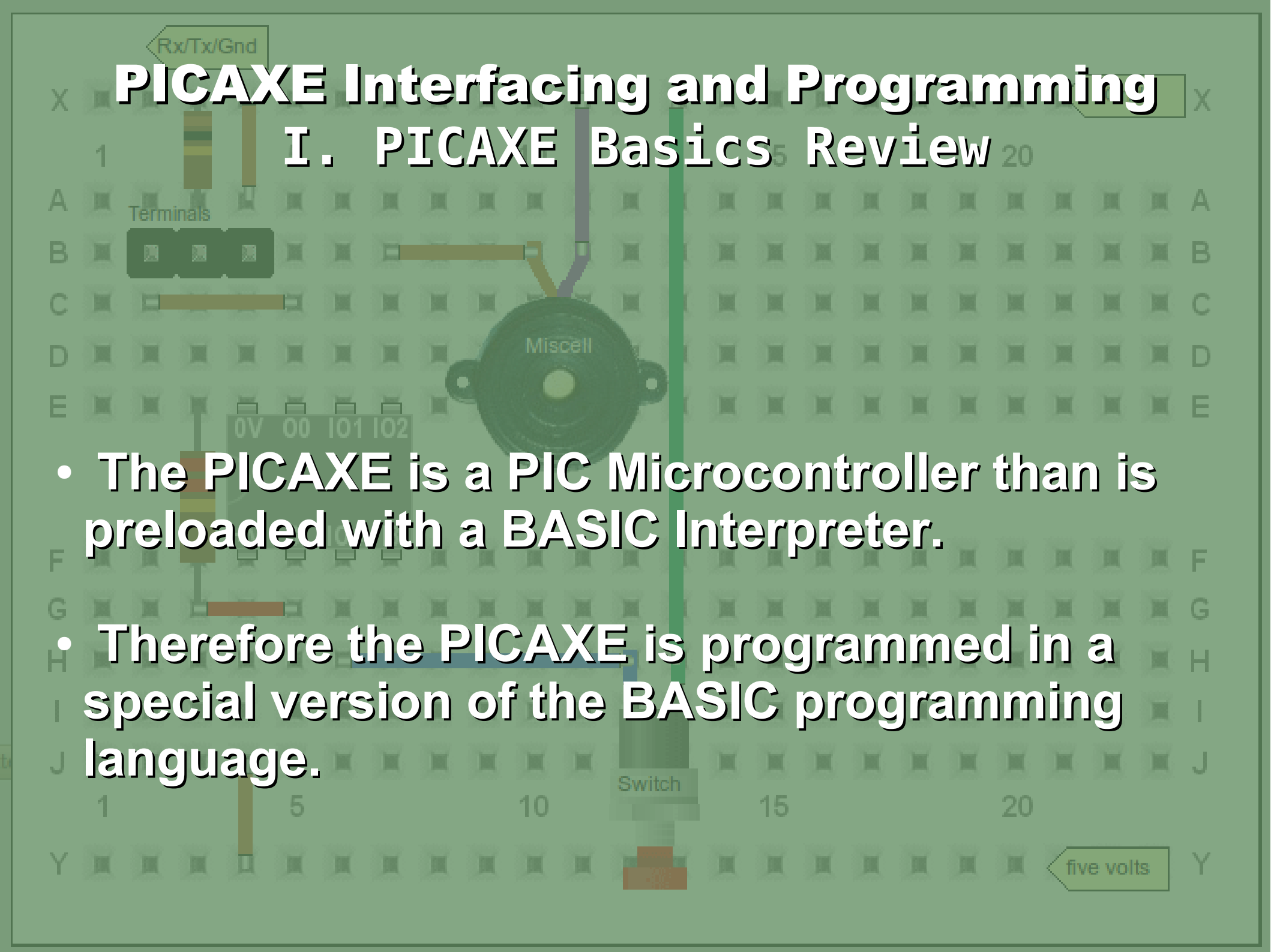
I. PICAXE BASICS REVIEW

PICAXE Interfacing and Programming
I. PICAXE Basics Review



- The PICAXE is a PIC Microcontroller than is preloaded with a BASIC Interpreter.
- Therefore the PICAXE is programmed in a special version of the BASIC programming language.

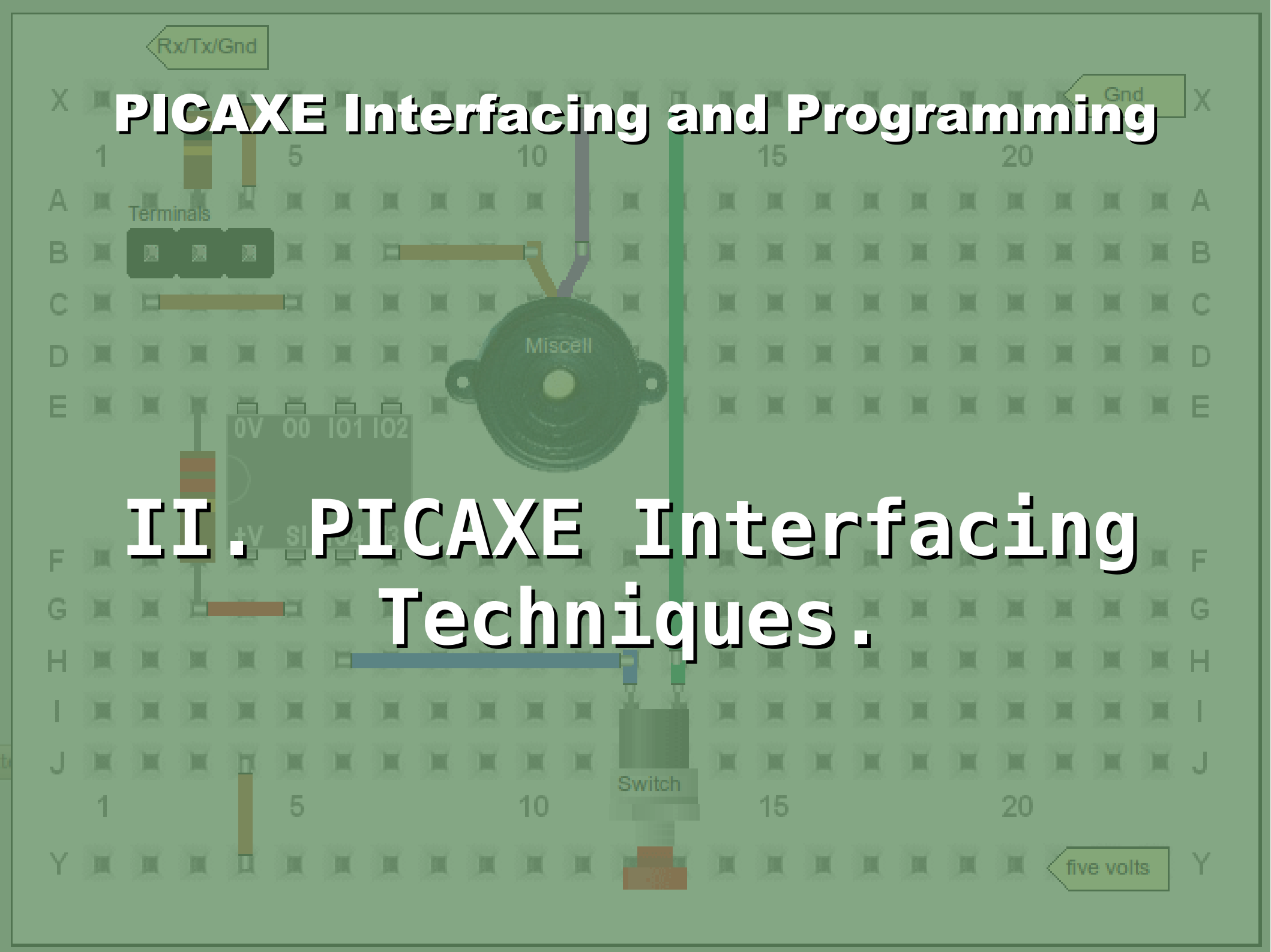
PICAXE Interfacing and Programming
I. PICAXE Basics Review



- The PICAXE is a PIC Microcontroller than is preloaded with a BASIC Interpreter.
- Therefore the PICAXE is programmed in a special version of the BASIC programming language.

PICAXE Interfacing and Programming
I. PICAXE Basics Review

- PICAXE Microcontroller, like the PIC Microcontrollers they are based upon, are available from 8 Pin to 40 Pin Devices in DIP and SMT packages.
- They are also available already preinstalled on solderable development boards.

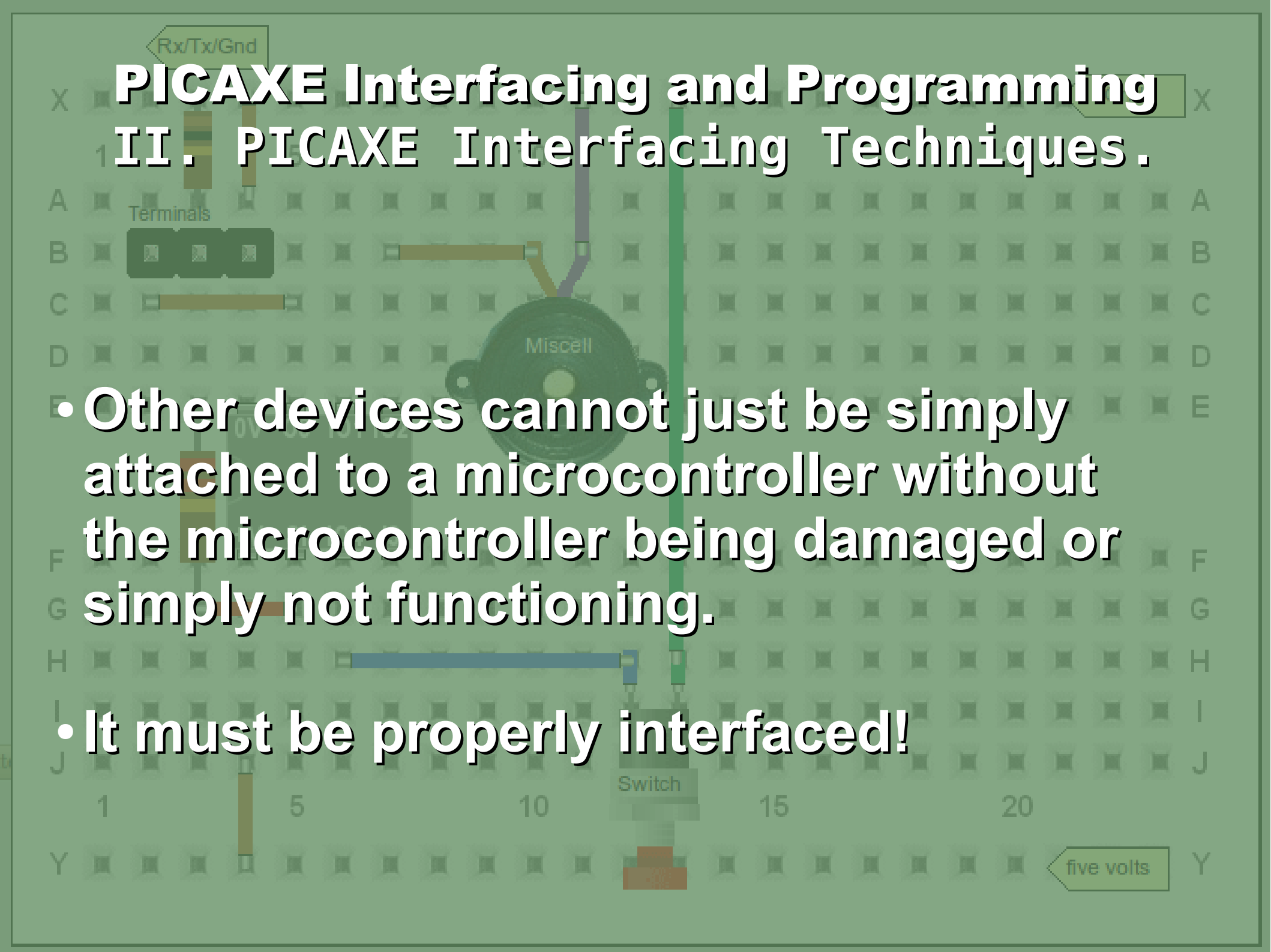


PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

PICAXE Interfacing and Programming

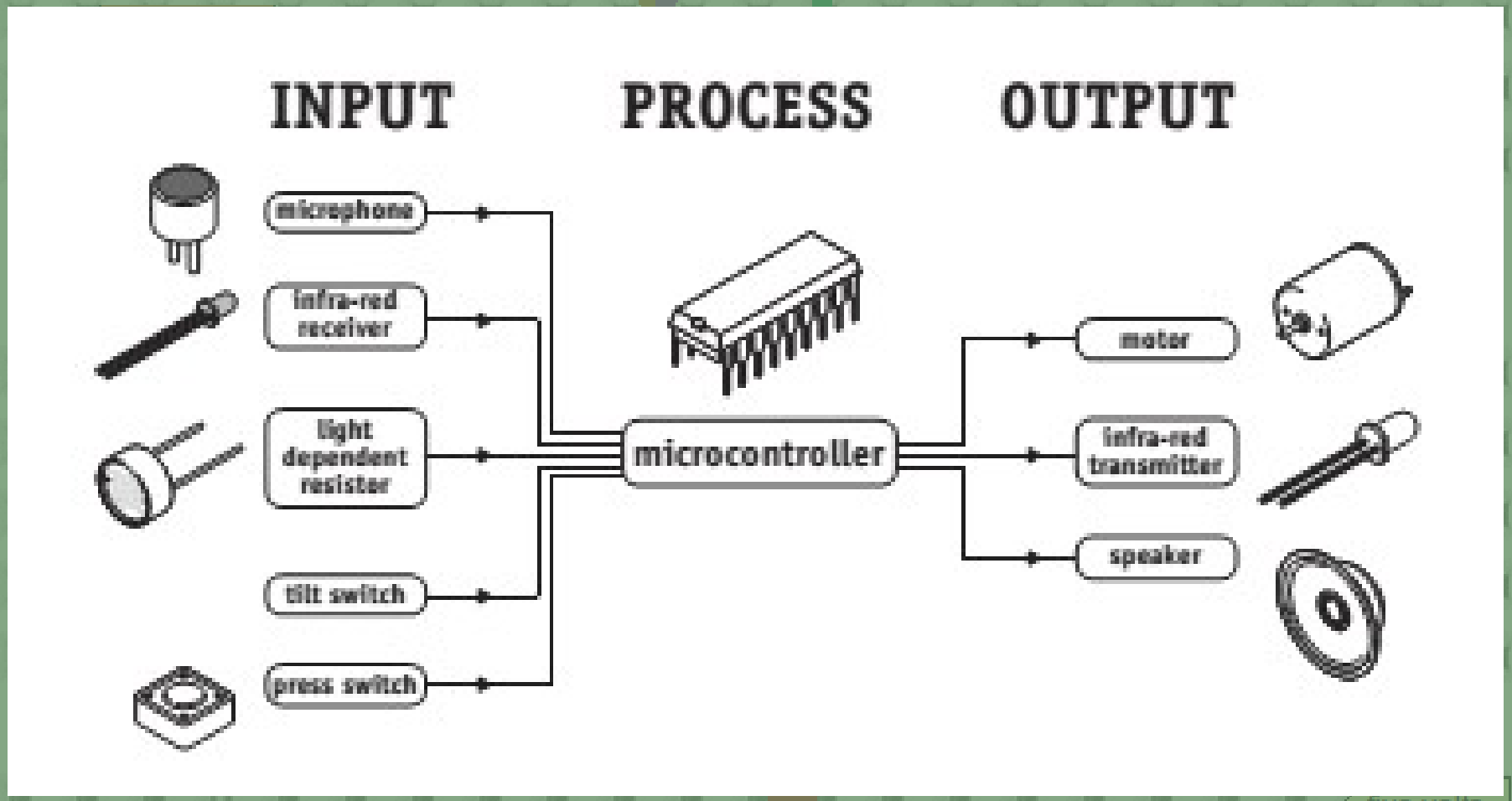
II. PICAXE Interfacing Techniques.



- Other devices cannot just be simply attached to a microcontroller without the microcontroller being damaged or simply not functioning.
- It must be properly interfaced!

PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

Microcontroller interfaces are in two general categories:

- **Output Interfaces**
- **Input Interfaces**
- **Advanced Interfaces (Input and Output combined)**

PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

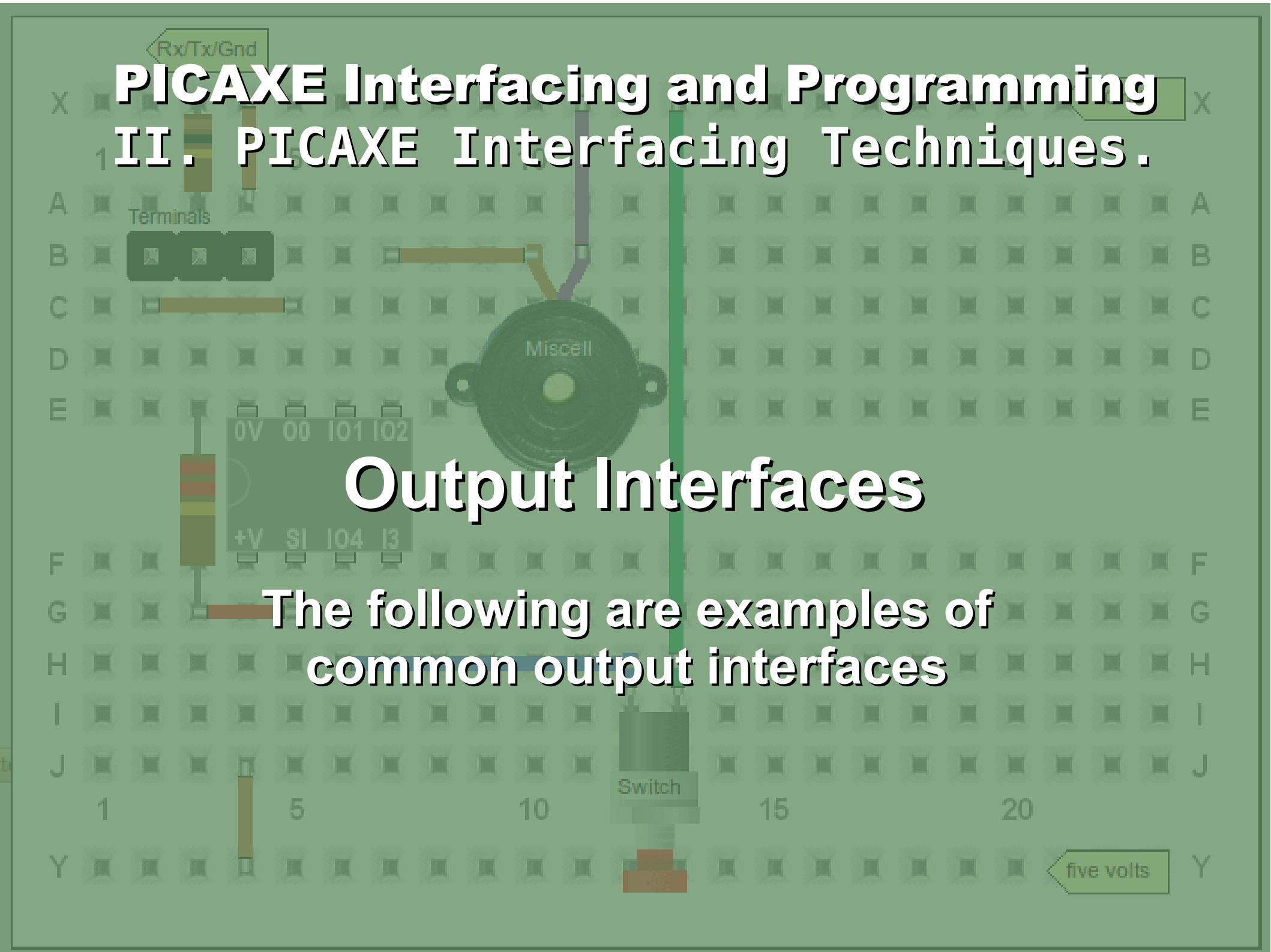
- These interface techniques are used with virtually all microcontroller platforms on the market today!

- These techniques are not all the techniques available but just the most popular ones.

- The interface illustrations used in these slides come out of the PICAXE manual #3 courtesy of Revolution Education PLC.

PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.



Output Interfaces

The following are examples of common output interfaces

PICAXE Interfacing and Programming

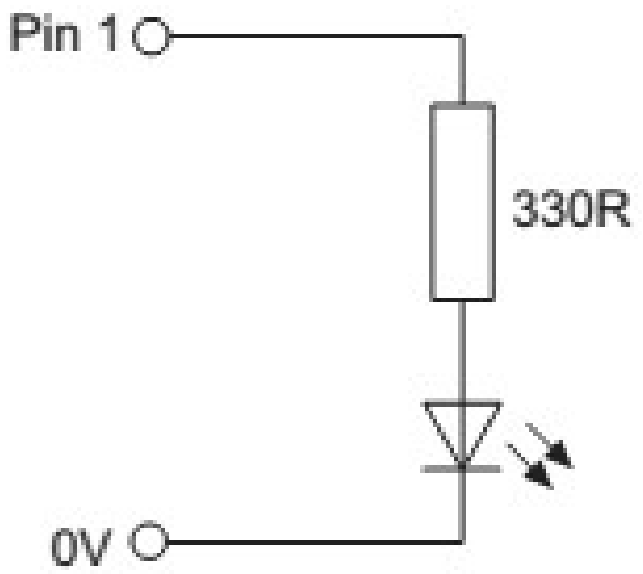
II. PICAXE Interfacing Techniques.

Output Technique #1: LED – High Active

LED connected to Ground Rail.

To switch on LED - `high 1`

To switch off LED - `low 1`



PICAXE Interfacing and Programming

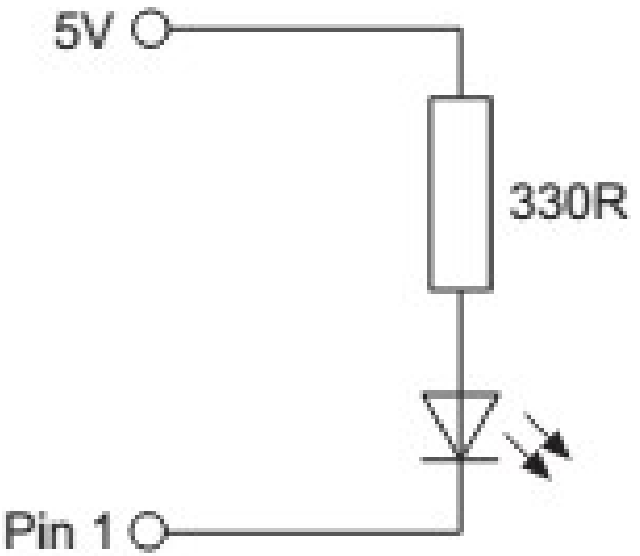
II. PICAXE Interfacing Techniques.

Output Technique #2: LED – Low Active

LED connected to Power Rail.

To switch on LED - `low 1`

To switch off LED - `high 1`



PICAXE Interfacing and Programming
II. PICAXE Interfacing Techniques.

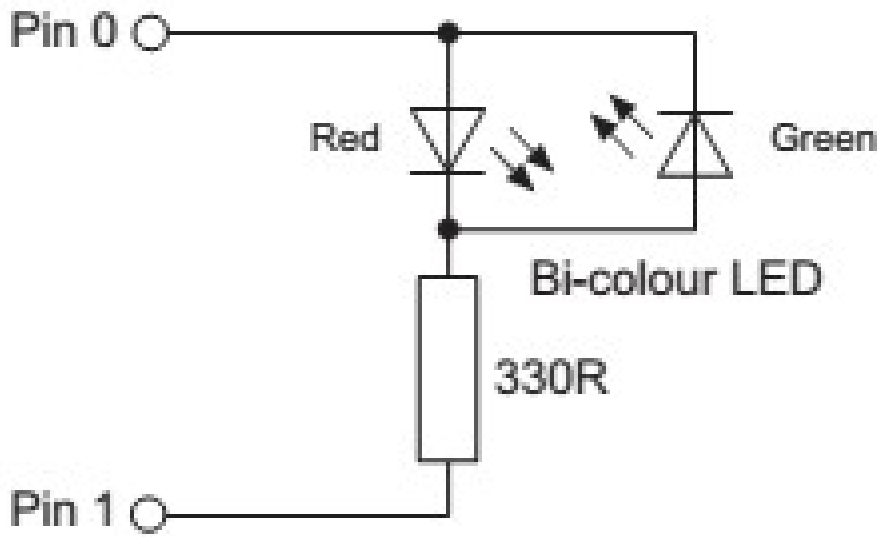
Output Technique #3:
Bi-Color LED

To switch on LED in red - high 0
low 1

To switch on LED in green - low 0
high 1

To switch off LED - low 0
low 1

or, high 0
high 1



PICAXE Interfacing and Programming

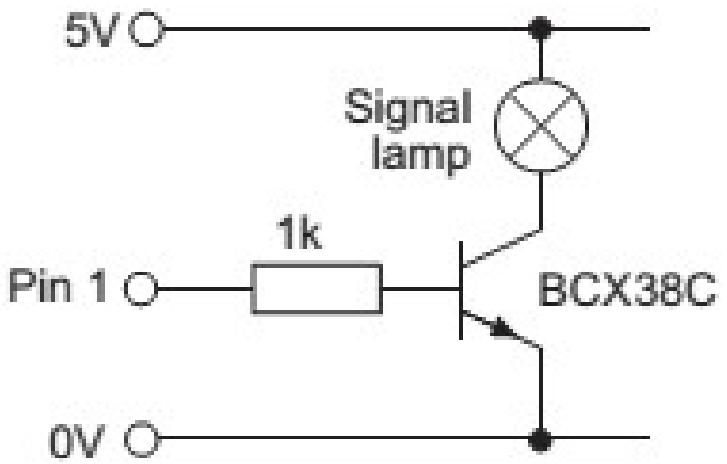
II. PICAXE Interfacing Techniques.

Output Technique #4: Lamp

To switch on Lamp - `high 1`

To switch off Lamp - `low 1`

signal lamp

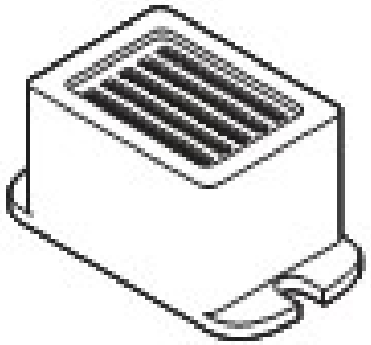


PICAXE Interfacing and Programming

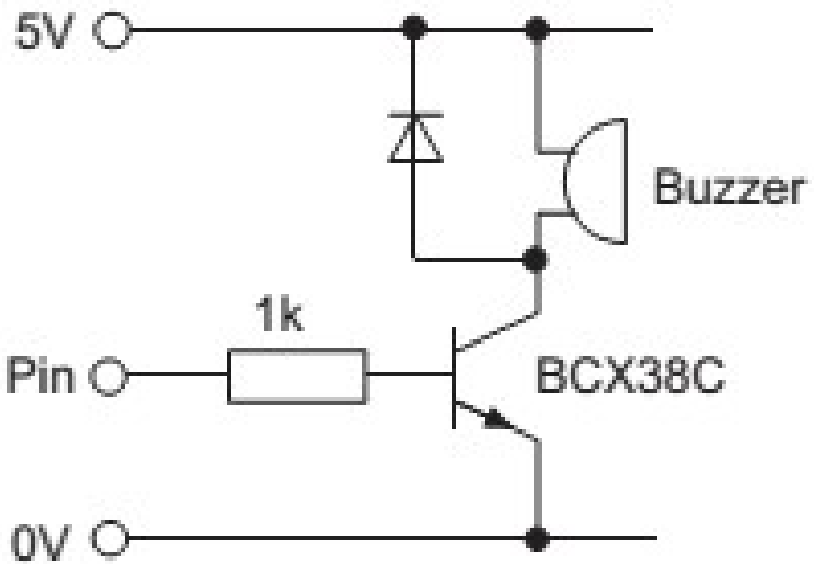
II. PICAXE Interfacing Techniques.

Output Technique #5: Sound with Buzzer

```
To switch on buzzer -    high 1
To switch off buzzer -   low 1
```



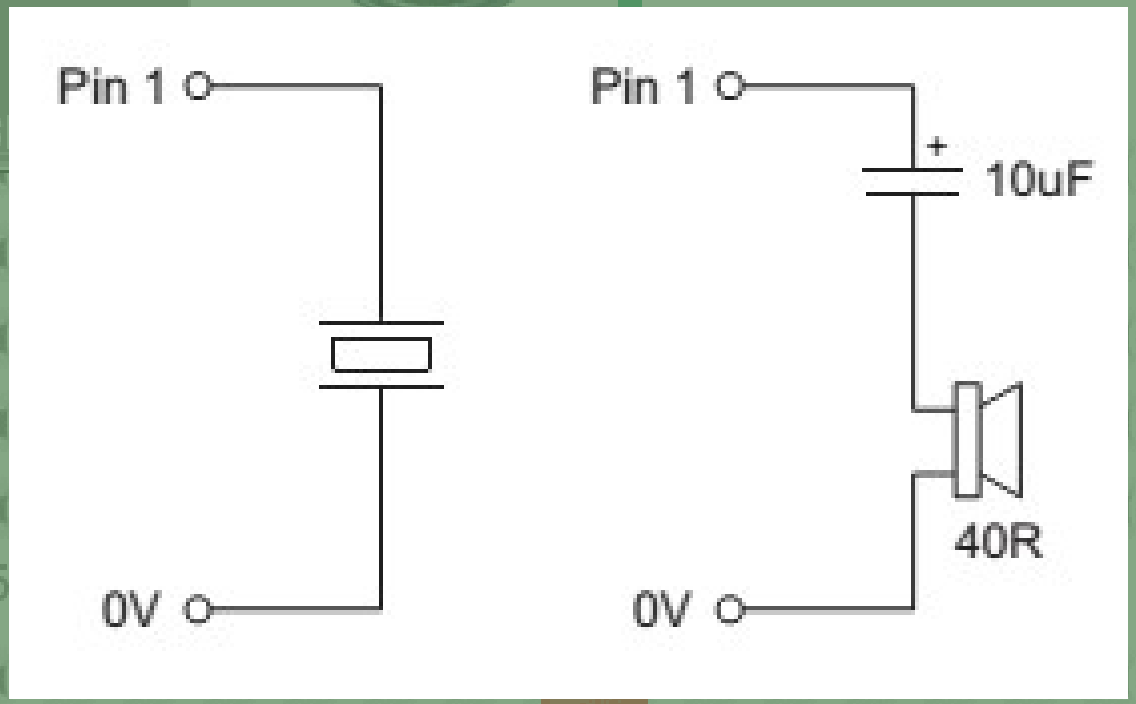
buzzer



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

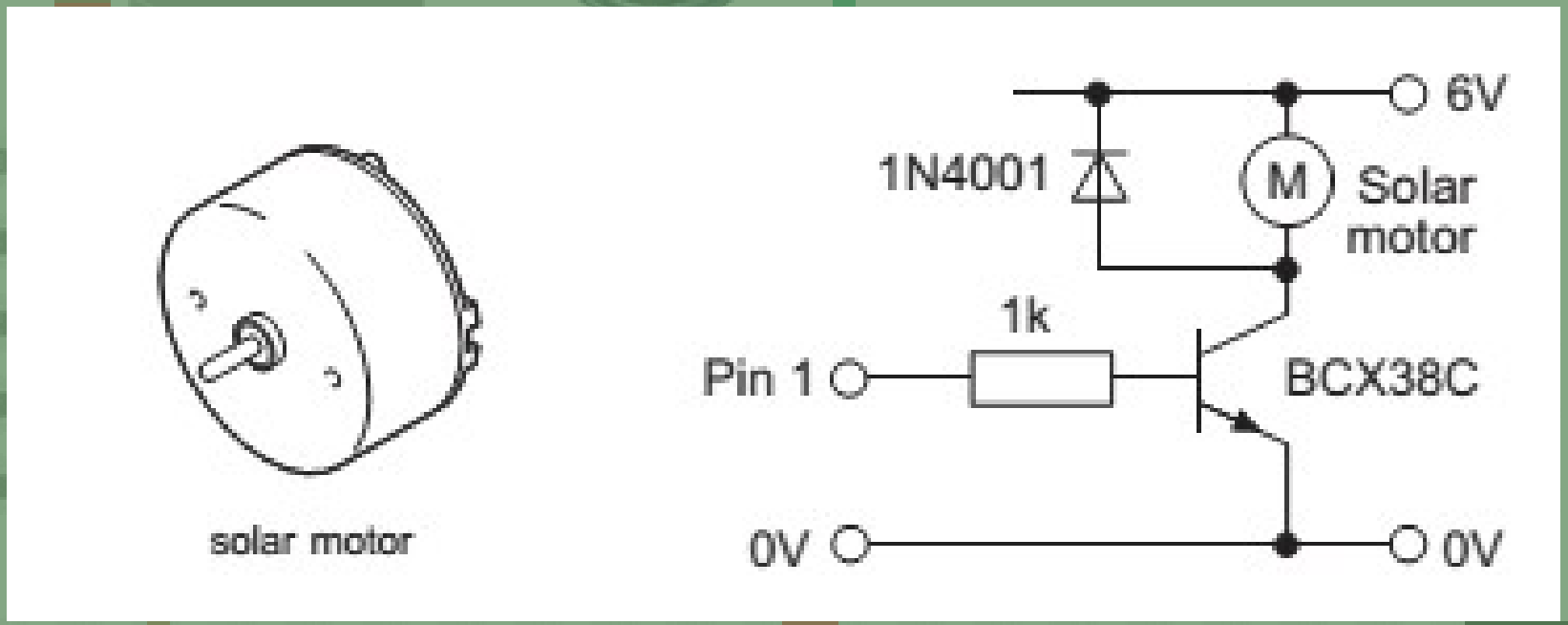
Output Technique #6: Sound Output with Piezo Buzzer or Speaker



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

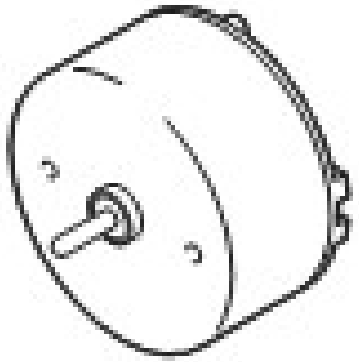
Output Technique #6: Sound with Piezo Buzzer or Speaker



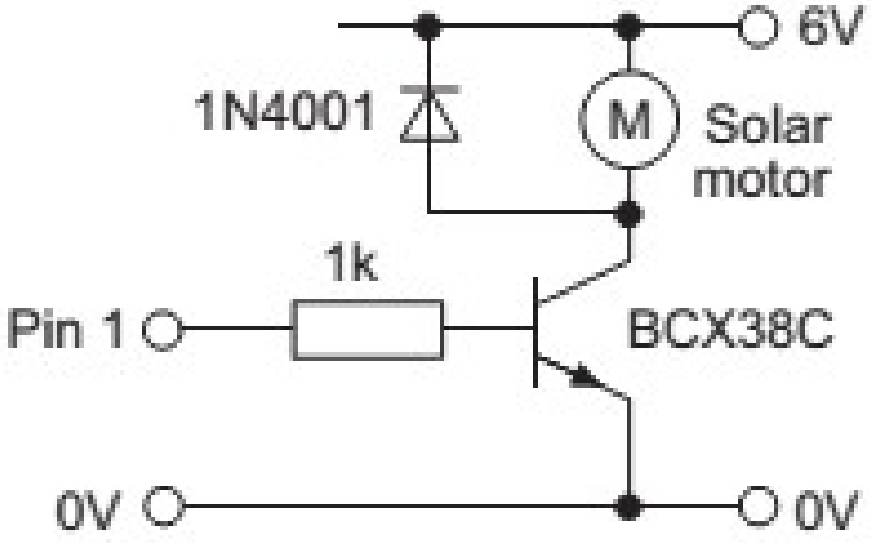
PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

Output Technique #7:
Turning a Motor On and Off – Light Duty



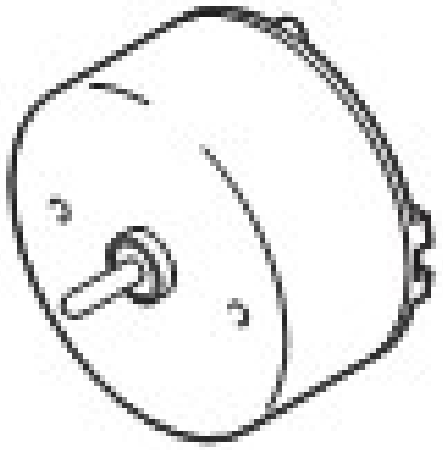
solar motor



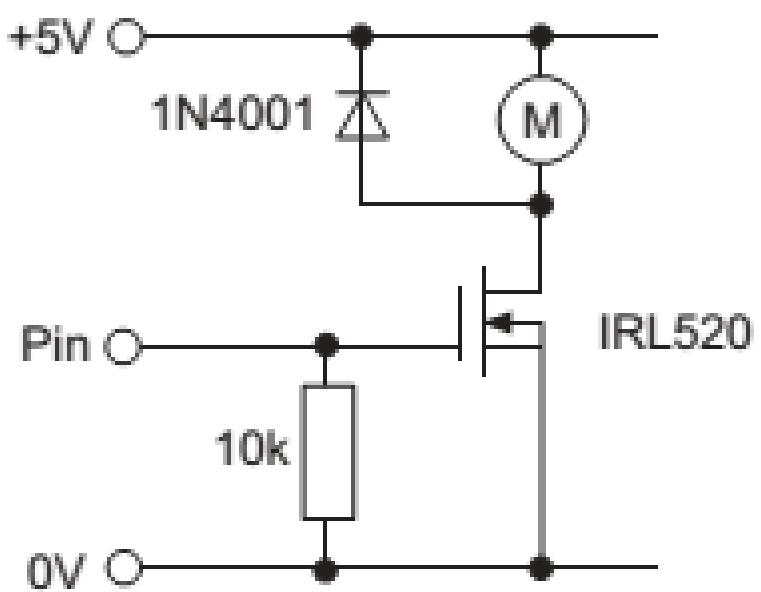
PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

Output Technique #8:
Turning a Motor On and Off – Heavy Duty



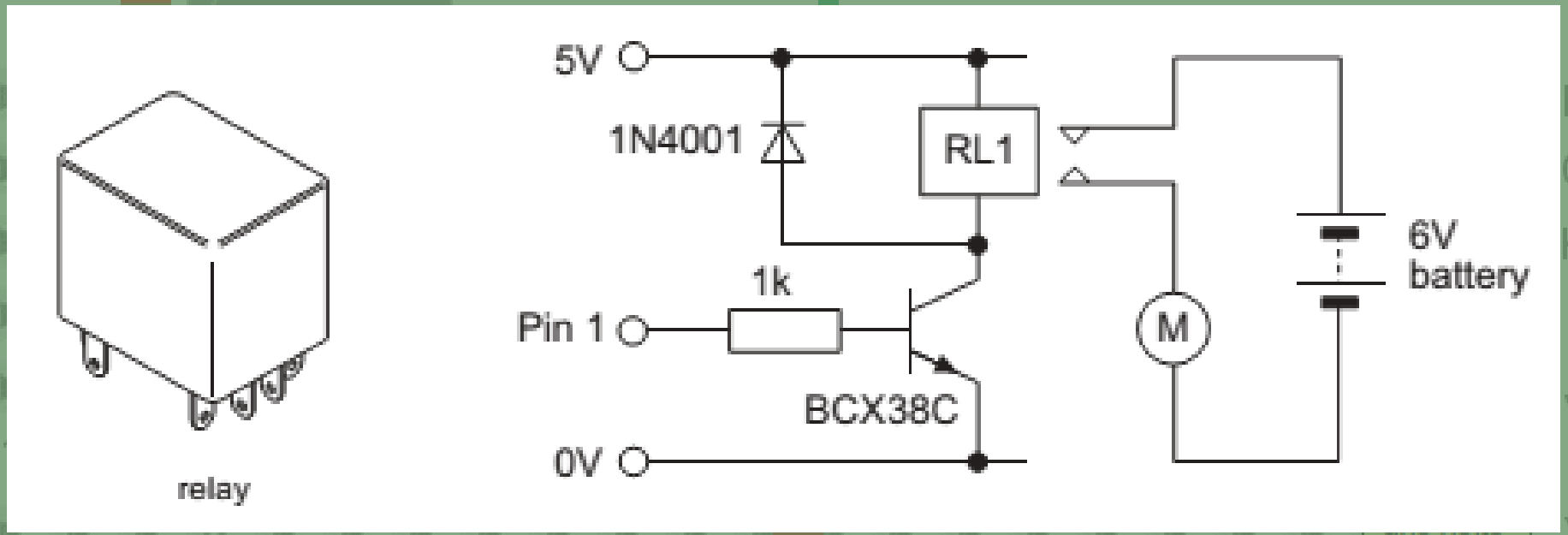
solar motor



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

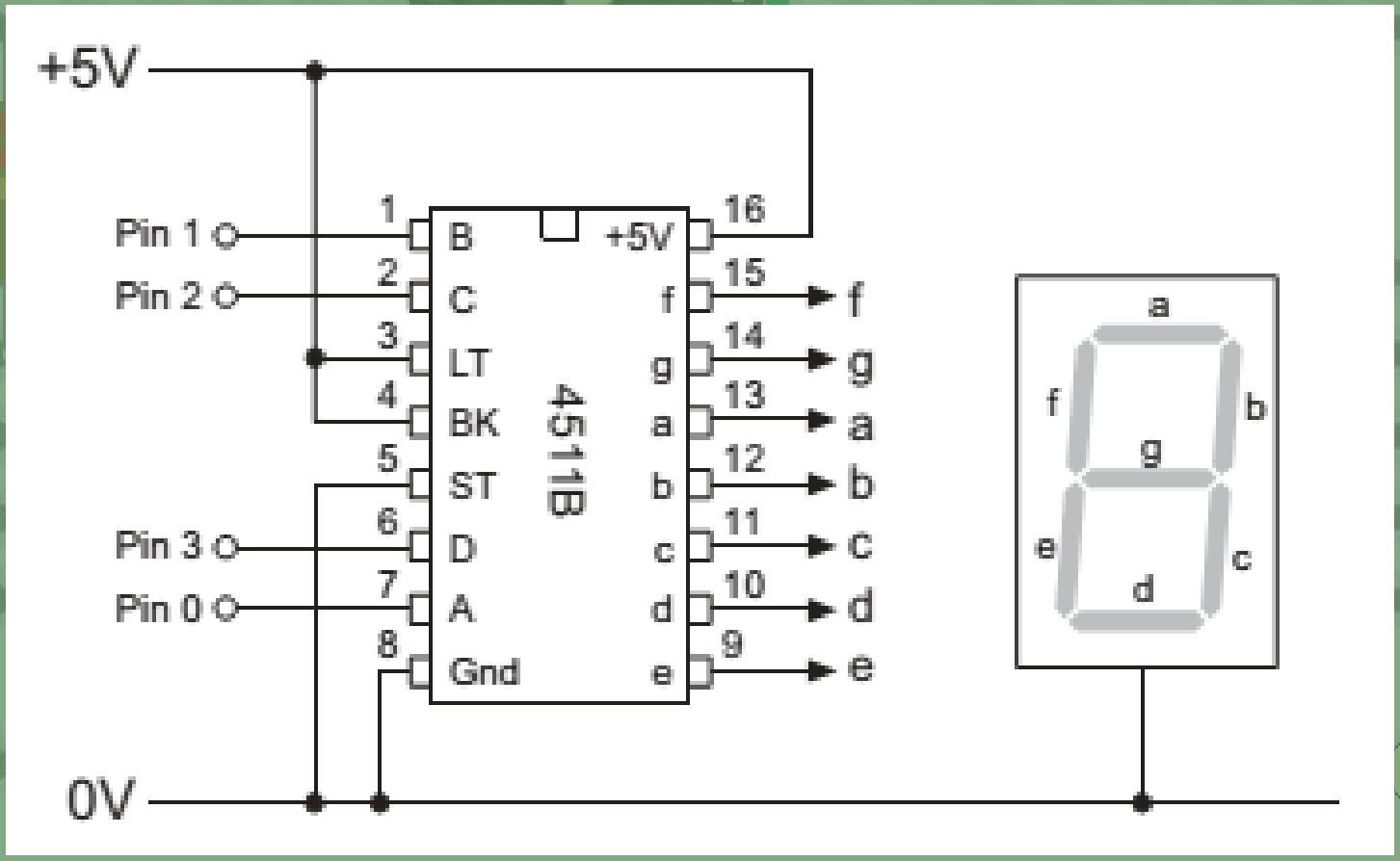
Output Technique #9: Turning on a Relay to Control loads.



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

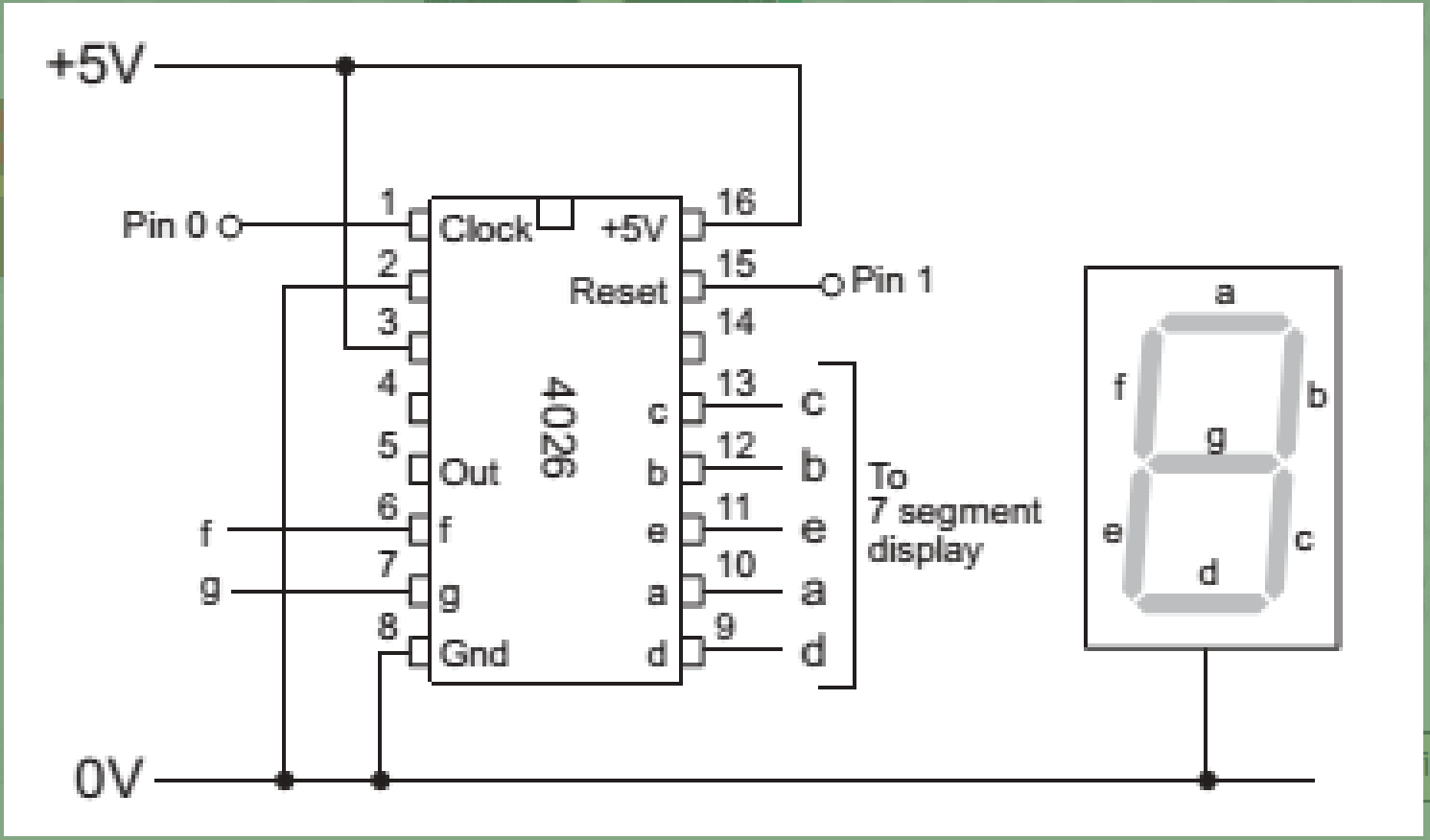
Output Technique #10:
Seven Segment LED Display: BCD Input.



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

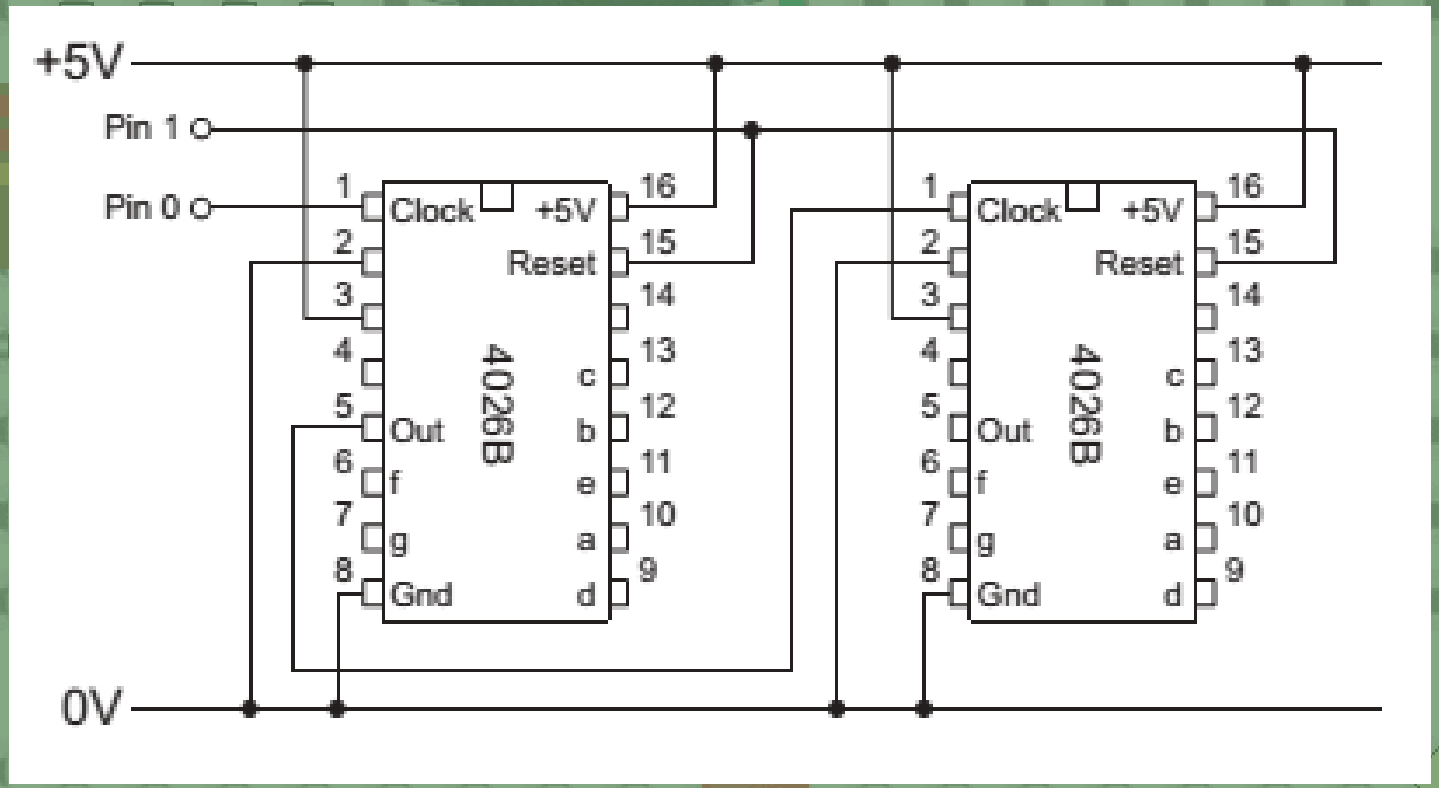
Output Technique #11:
Seven Segment LED Display: Clock.



PICAXE Interfacing and Programming

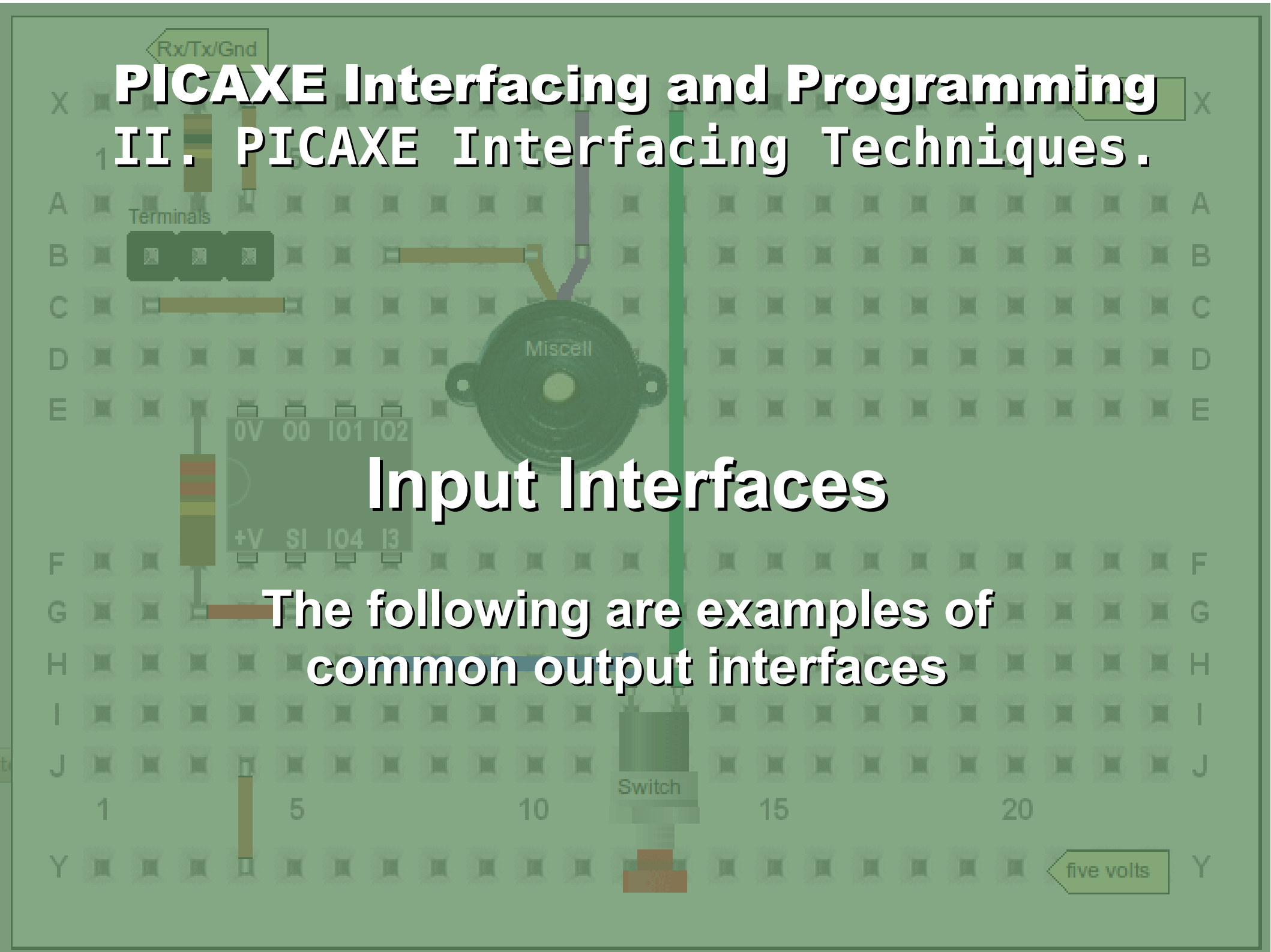
II. PICAXE Interfacing Techniques.

Output Technique #11:
Two 7 Segment LED Display: Clock.



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.



Input Interfaces

The following are examples of common output interfaces

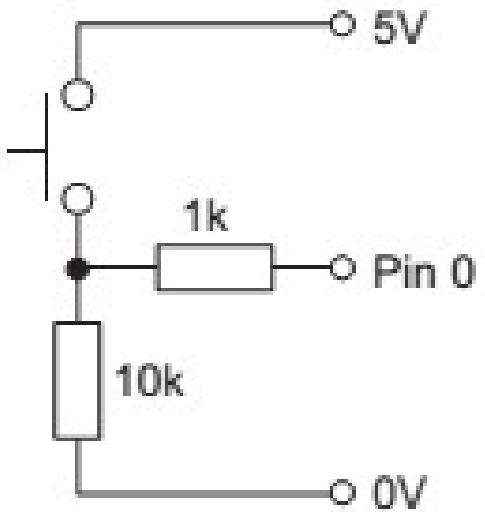
PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

Input Technique #1: Switch – Active High

Goto 'jump' when switch is open: if pin0 = 0 then jump

Goto 'jump' when switch is closed: if pin0 = 1 then jump



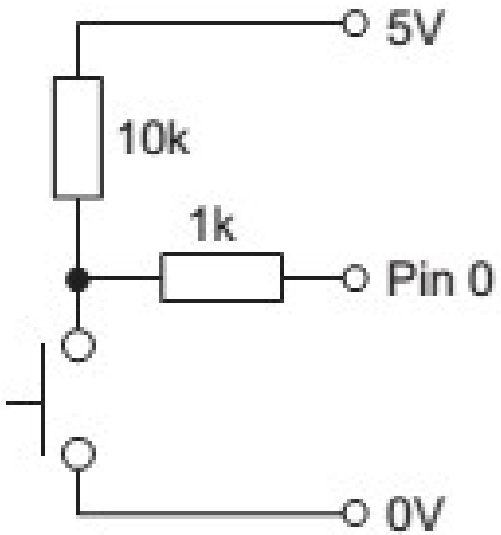
PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

Input Technique #2: Read a Switch – Active Low

Goto 'jump' when switch is open: if pin0 = 1 then jump

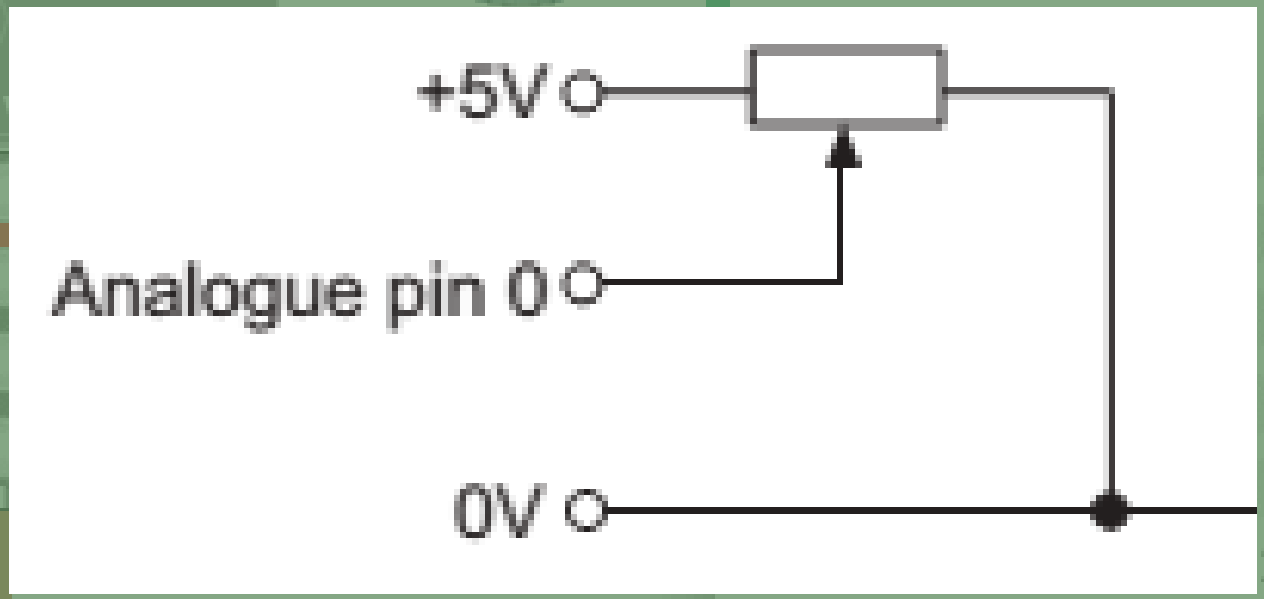
Goto 'jump' when switch is closed: if pin0 = 0 then jump



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

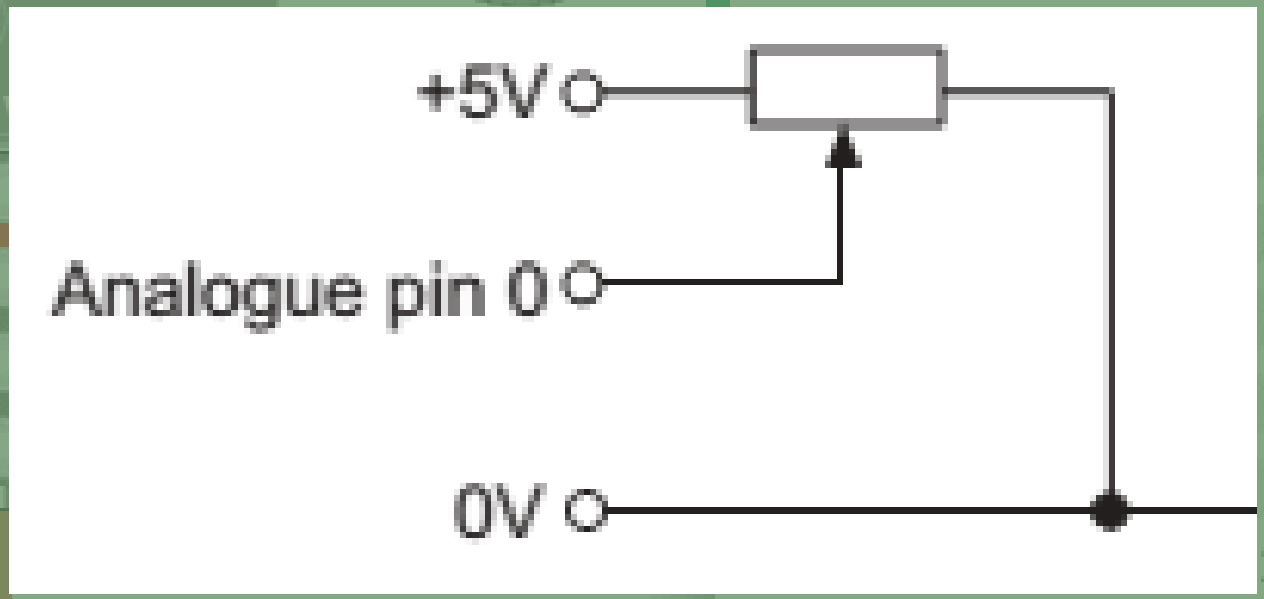
Input Technique #3: Analog Voltage (0-5 VDC)



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

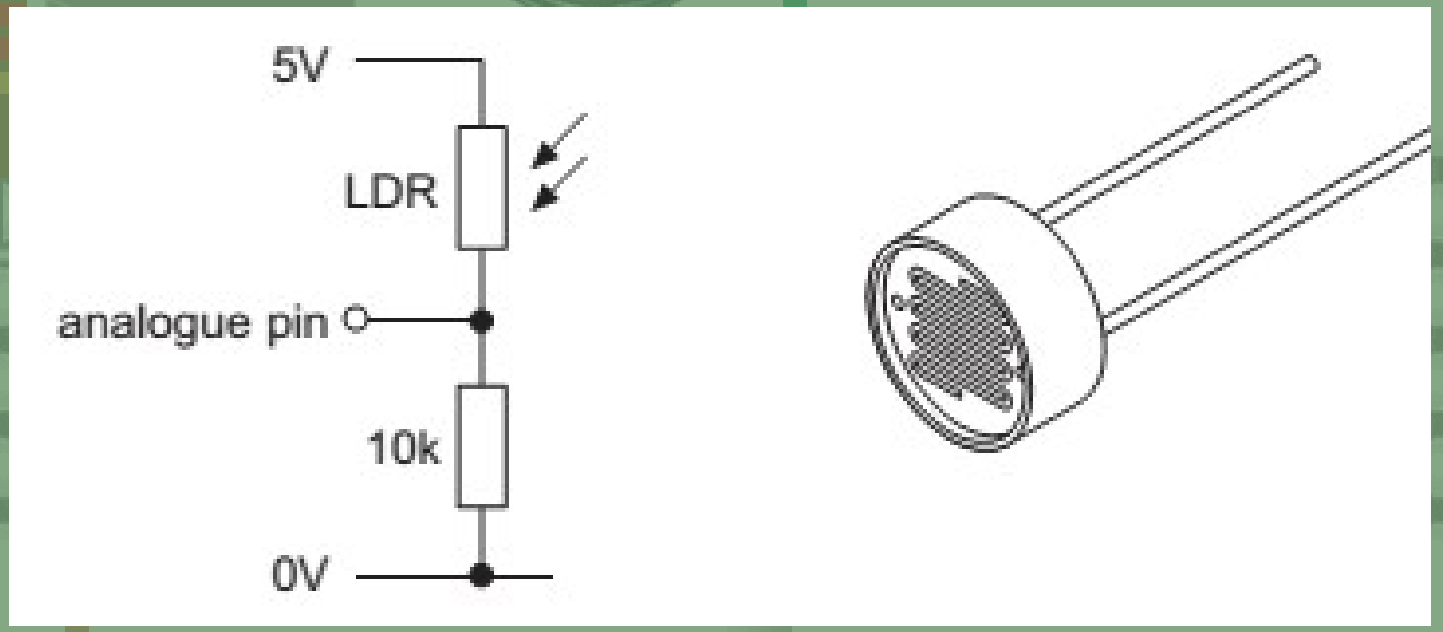
Input Technique #3: Analog Voltage (0-5 VDC)



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

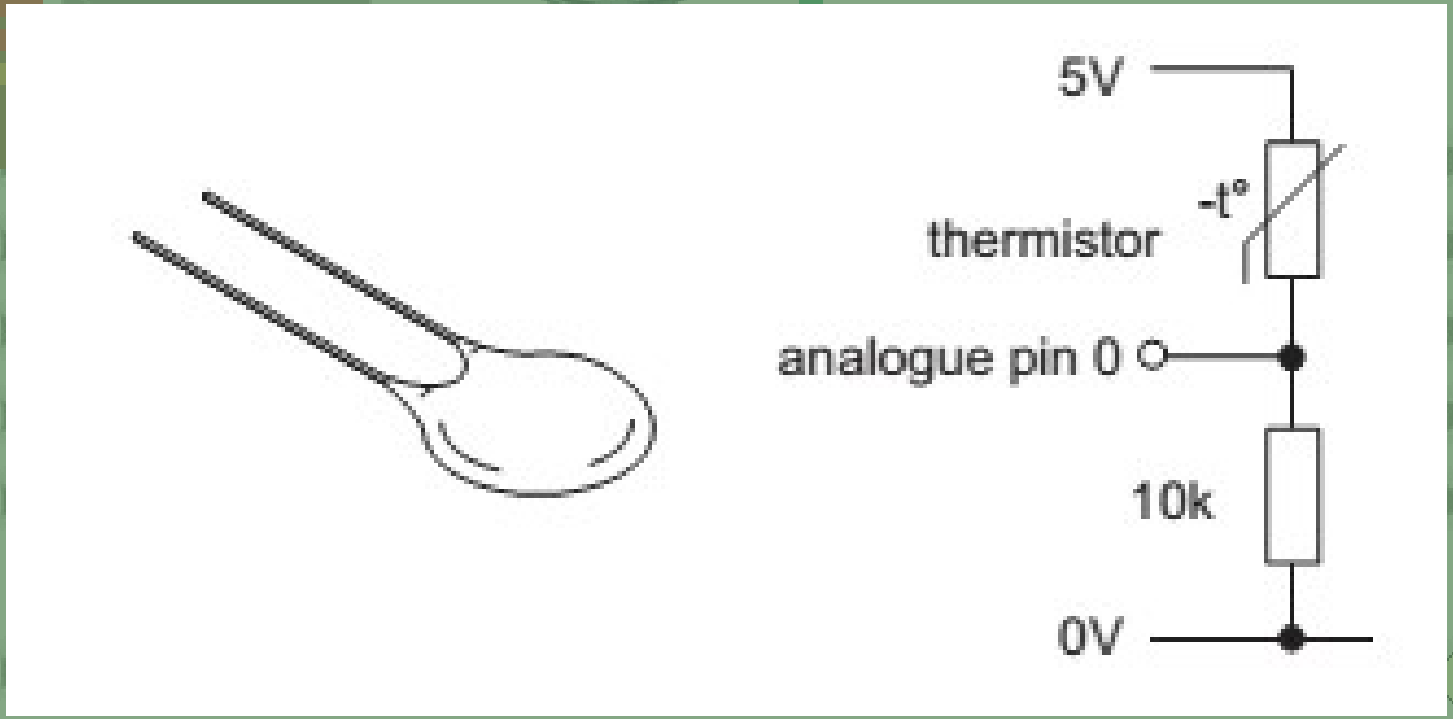
Input Technique #4:
Light Sensor (CdS Cell - Photoresistor)



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

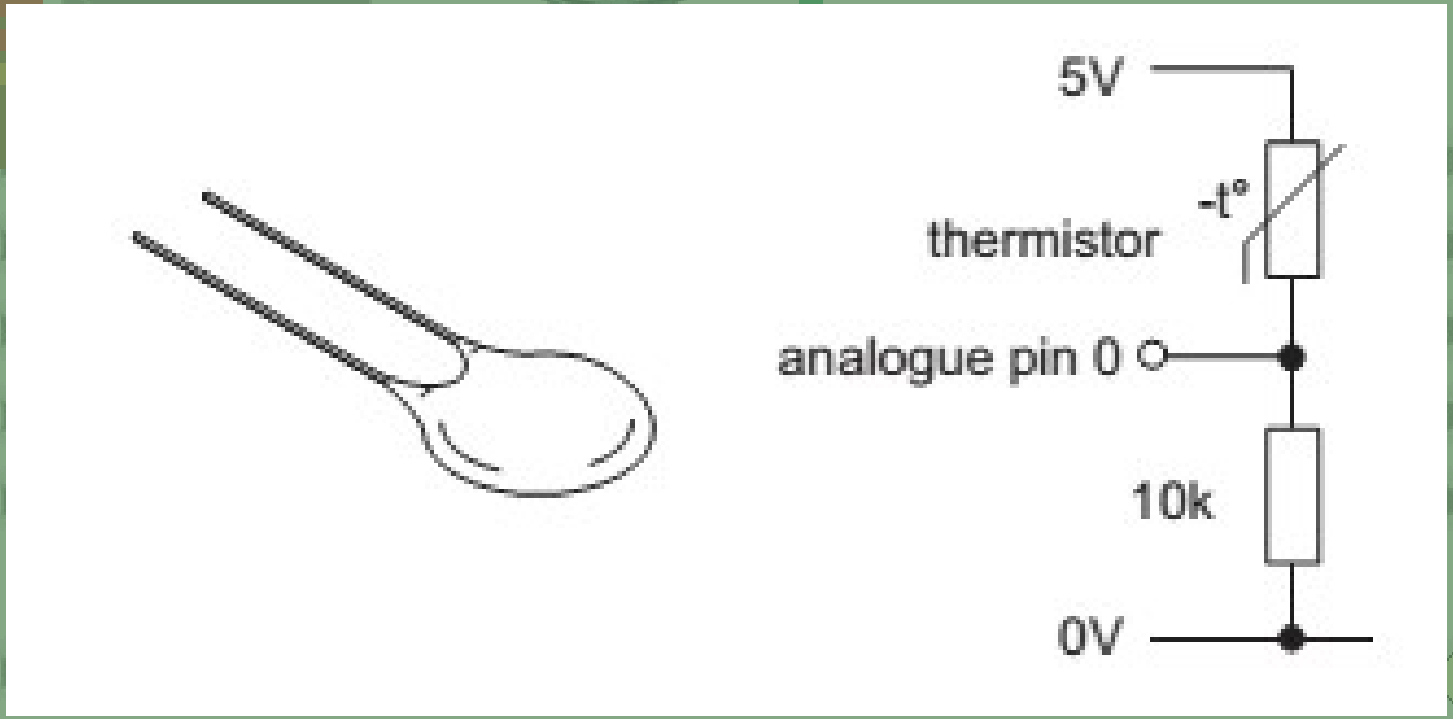
Input Technique #5:
Tempature Sensor (Thermistor)



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

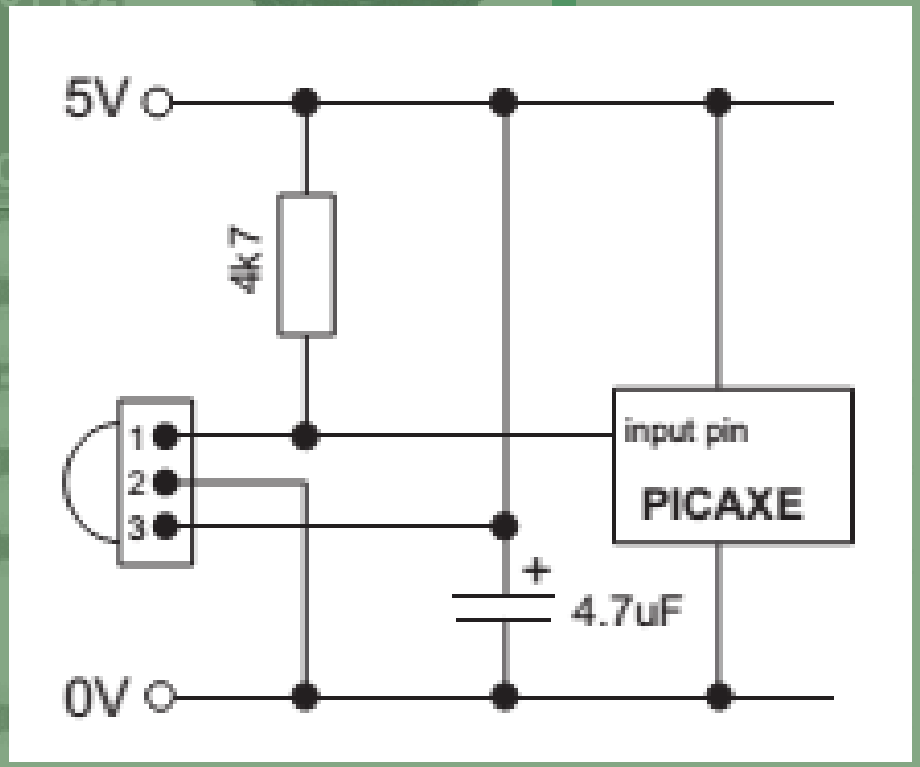
Input Technique #5:
Tempature Sensor (Thermistor)



PICAXE Interfacing and Programming

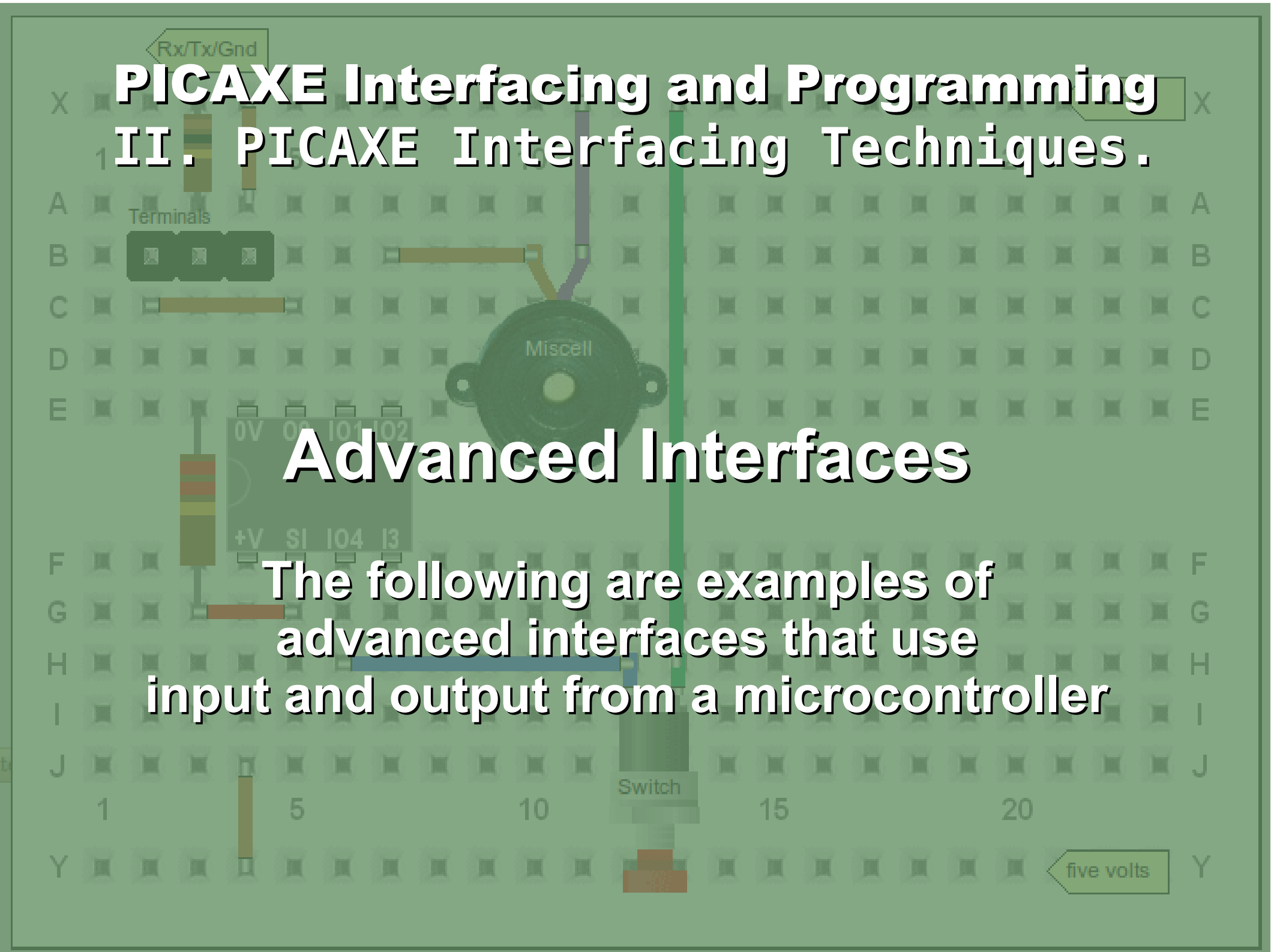
II. PICAXE Interfacing Techniques.

Input Technique #6: Infrared Receiver



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.



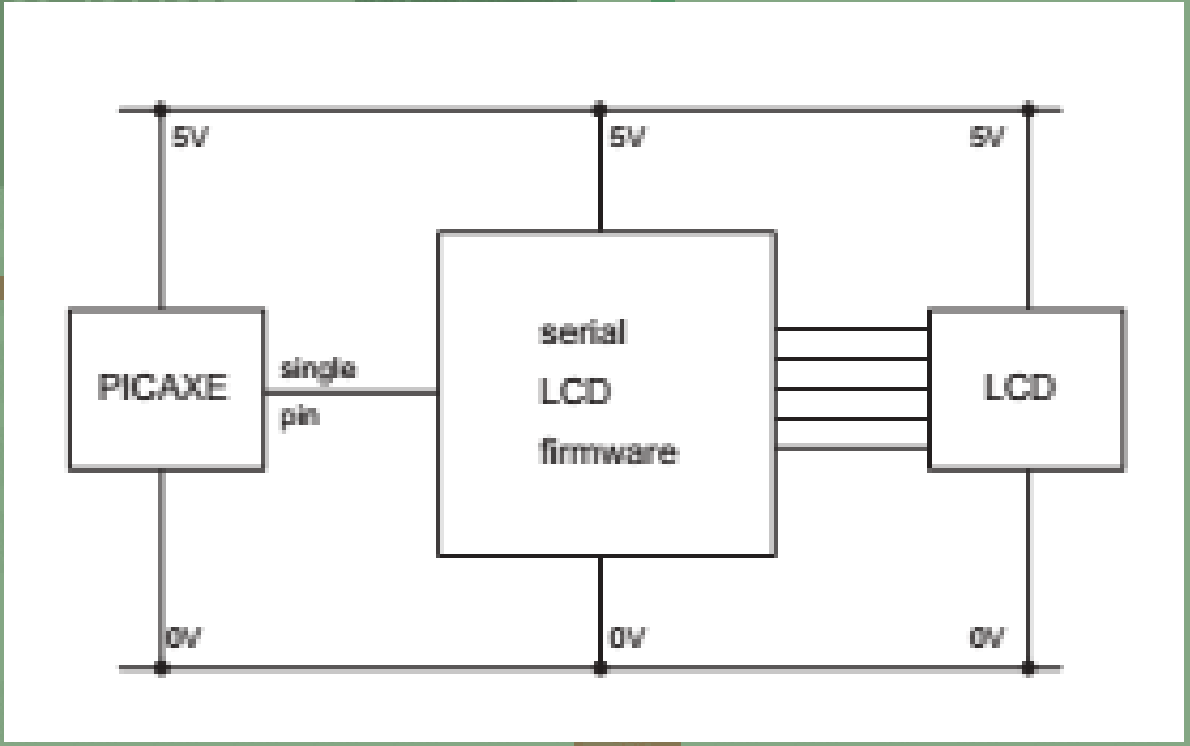
Advanced Interfaces

The following are examples of advanced interfaces that use input and output from a microcontroller

PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

Advanced Technique #1: Serial Character LCD Display Driver IC Block Diagram



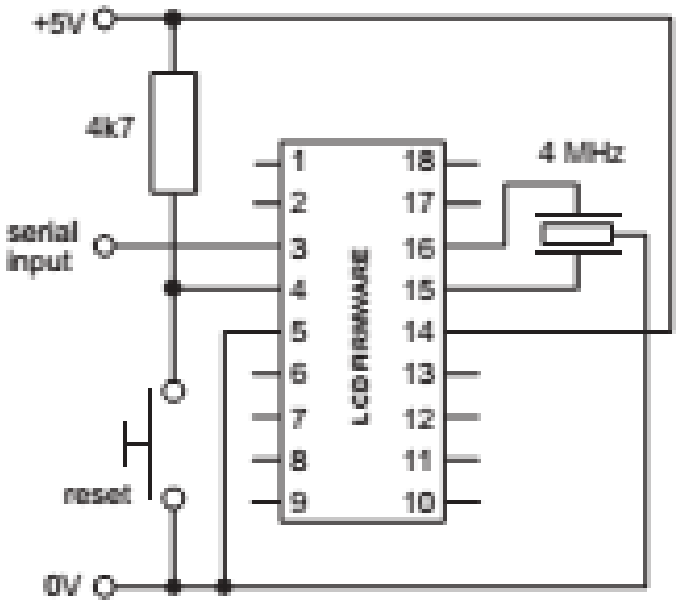
PICAXE Interfacing and Programming

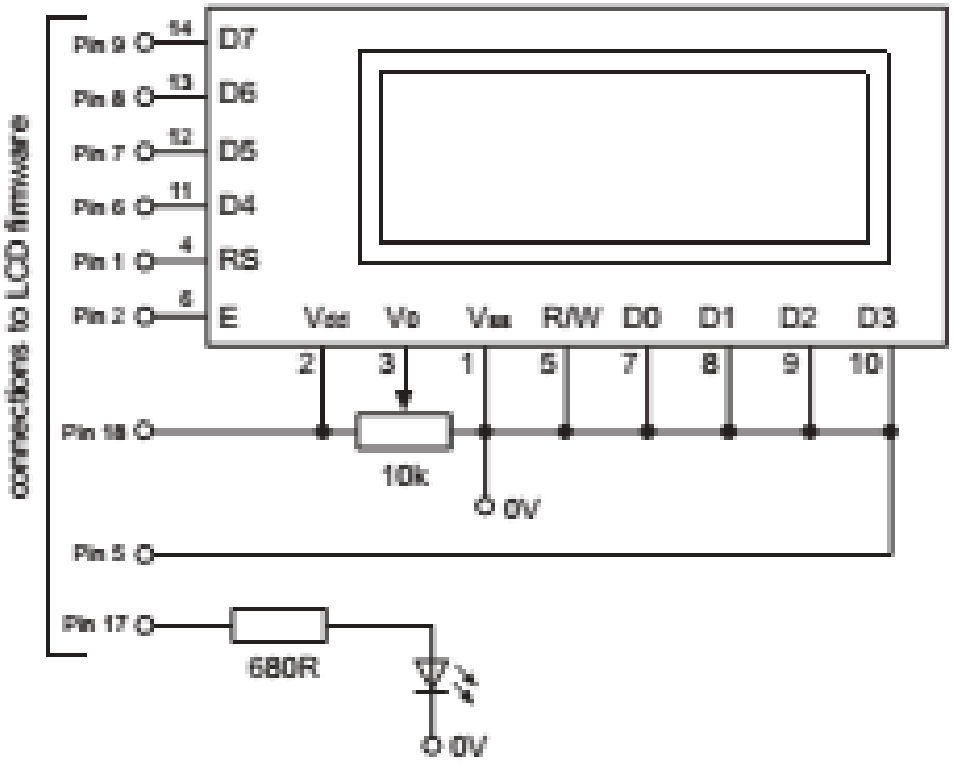
II. PICAXE Interfacing Techniques.

Advanced Technique #1:

Serial Character LCD Display Driver IC

Schematic Diagram

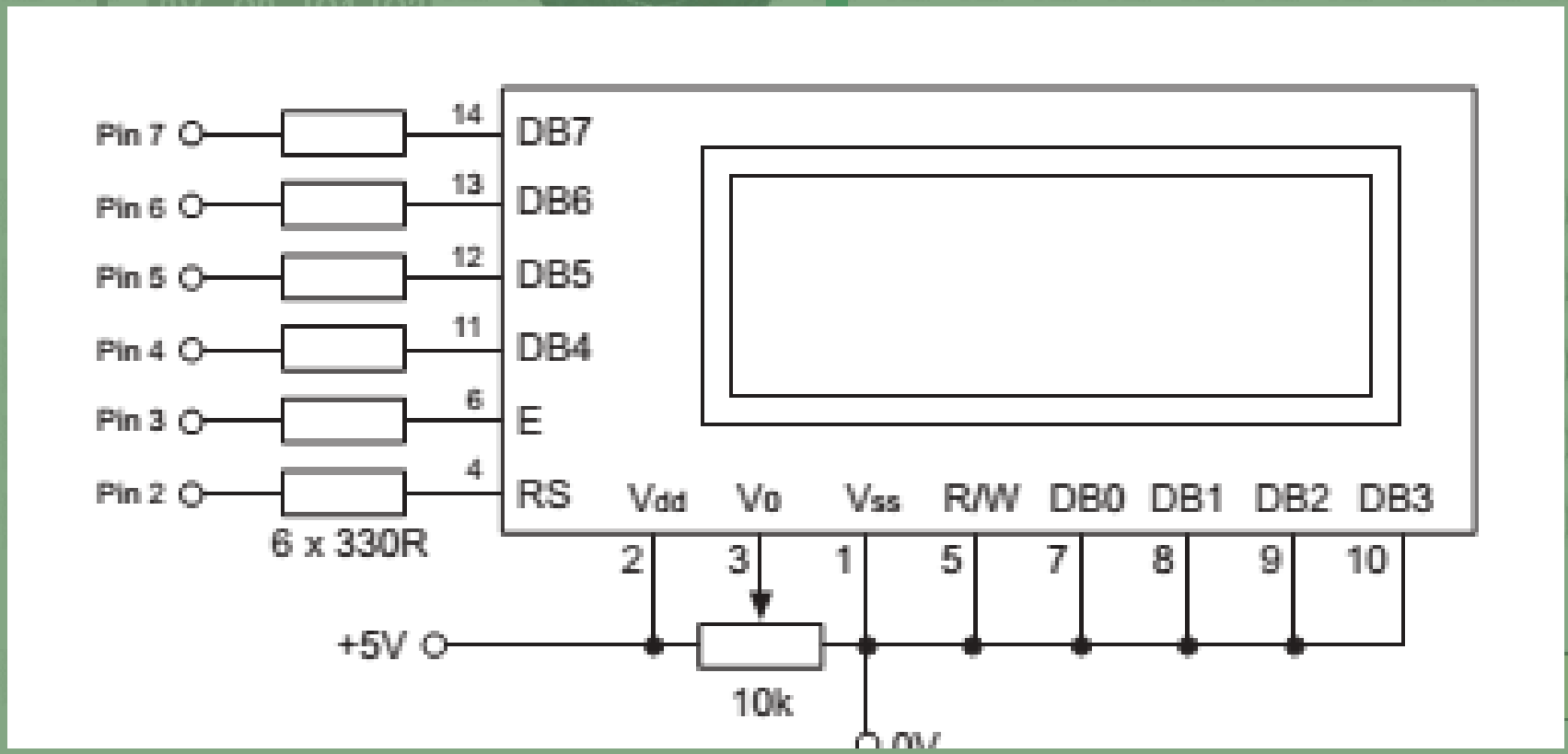




PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

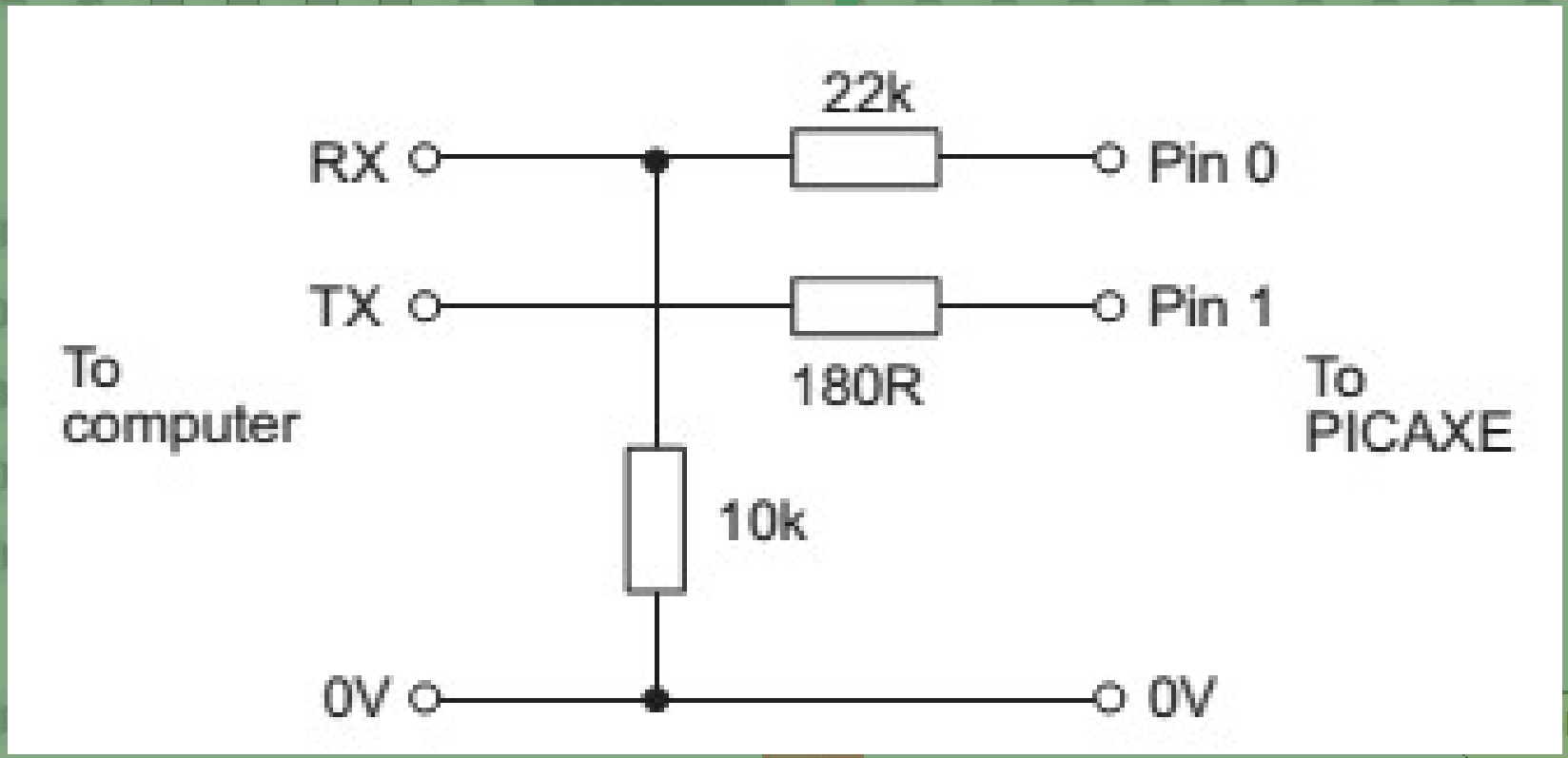
Advanced Technique #2:
Serial Character LCD Display Module



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

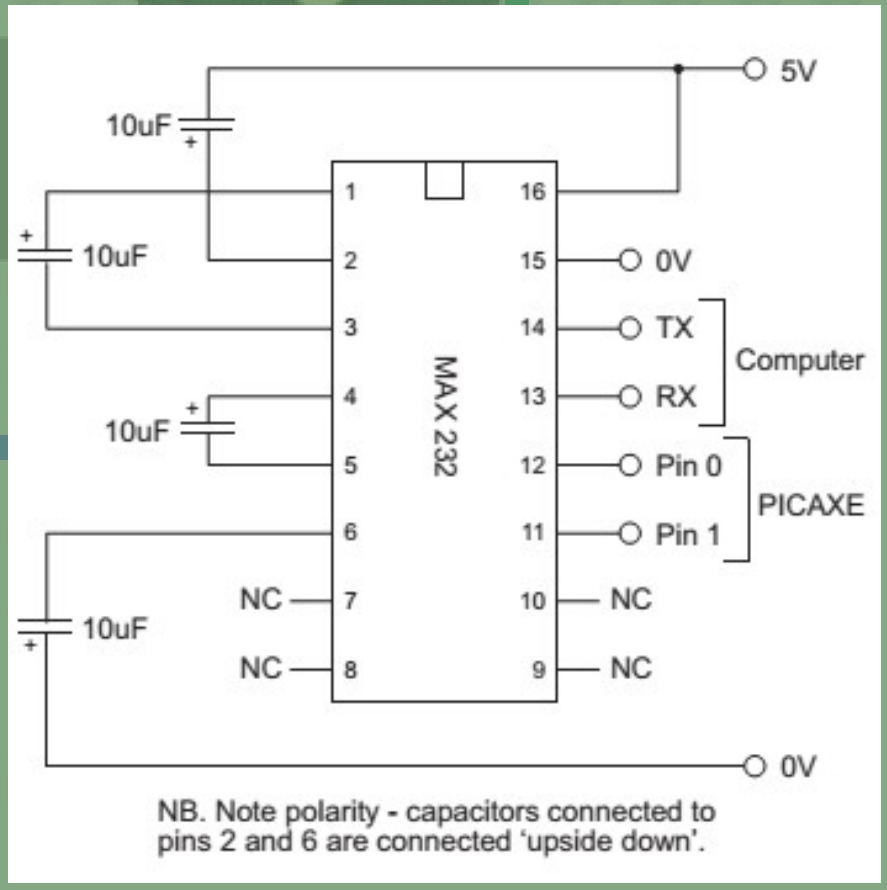
Advanced Technique #3:
Serial Port (RS232) Interface - Direct



PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

Advanced Technique #4: Serial Port (RS232) Interface - IC



NB. Note polarity - capacitors connected to pins 2 and 6 are connected 'upside down'.

PICAXE Interfacing and Programming
II. PICAXE Interfacing Techniques.

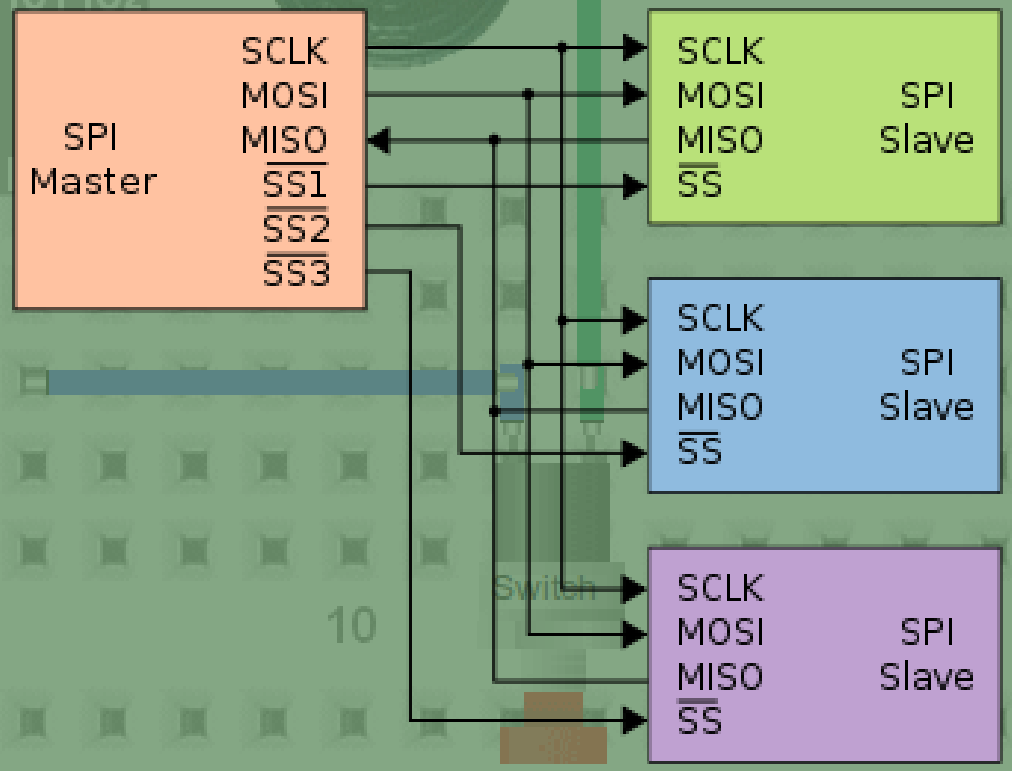
Advanced Technique #5:
SPI – Serial Peripheral Interface

- Developed by Motorola: Now Freescale Semi.
- Master-Slave Configuration
- Synchronous Communicaion: Clock Signal
- Four Wire Serial Bus: Clock, MOSI, MISO, Select
- Faster than I2C
- https://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus

PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

Advanced Technique #5:
SPI – Serial Peripheral Interface



PICAXE Interfacing and Programming
II. PICAXE Interfacing Techniques.

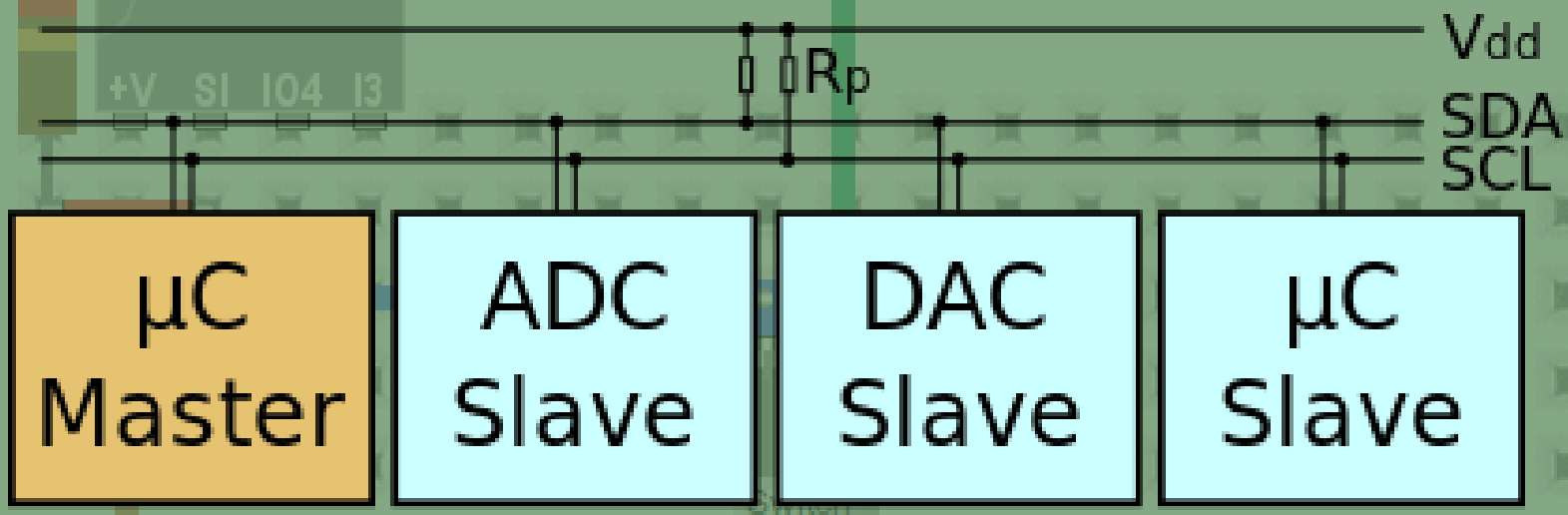
Advanced Technique #6:
I2C – Inter Integrated Circuit

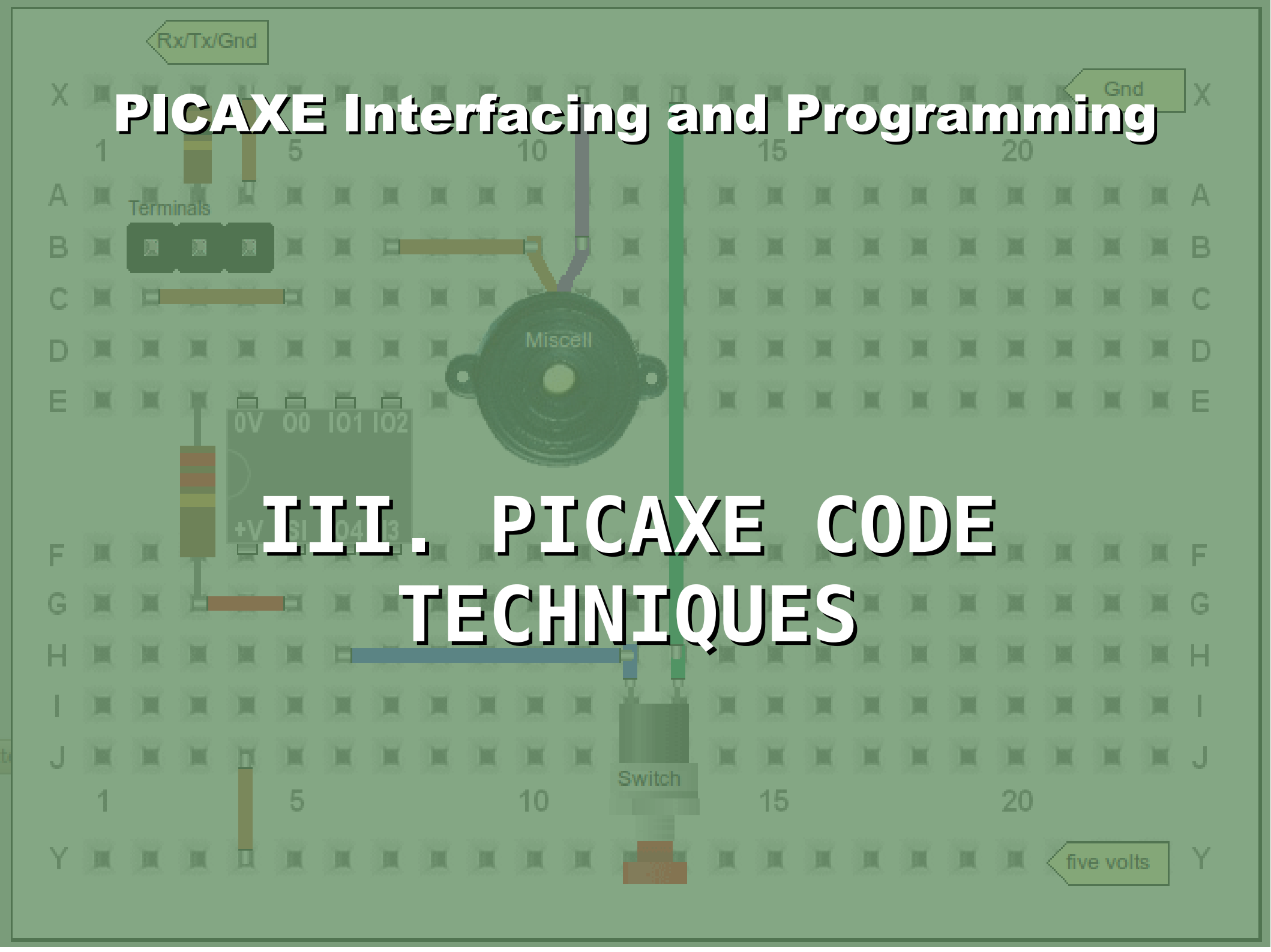
- Developed by Phillips in 1982: Now NXP Semi.
- Master-Slave Configuration
- Asynchronous Communication: Sync with Data
- Two Wire Serial Standard: SCK, SDL
- Slower then SPI but less pins
- https://en.wikipedia.org/wiki/I%C2%B2C

PICAXE Interfacing and Programming

II. PICAXE Interfacing Techniques.

Advanced Technique #6:
I2C – Inter Integrated Circuit





PICAXE Interfacing and Programming

III. PICAXE CODE TECHNIQUES

PICAXE Interfacing and Programming

III. PICAXE Code Techniques.

PICAXE MANUALS

Free for download, the PICAXE Manuals are well laid out and well document how to get started and use the PICAXE Microcontroller.

PICAXE Interfacing and Programming
III. PICAXE Code Techniques.

PICAXE MANUALS

- Manual 1: Getting Started and Products
- Manual 2: Programming Commands
- Manual 3: Interfacing Techniques
- Manual 4: Programming Editor
- Manual 5: Blocky
  Visual Programming Editor

PICAXE Interfacing and Programming
III. PICAXE Code Techniques.

- Many C++ based microcontroller environments require the program to write (or use others) objects (functions) in order to easily use external devices.
- The PICAXE environment made this unnecessary as many useful commands are already included in the interpreter.

PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

- Now we will show you some basic techniques for accomplishing useful tasks in your own programs.

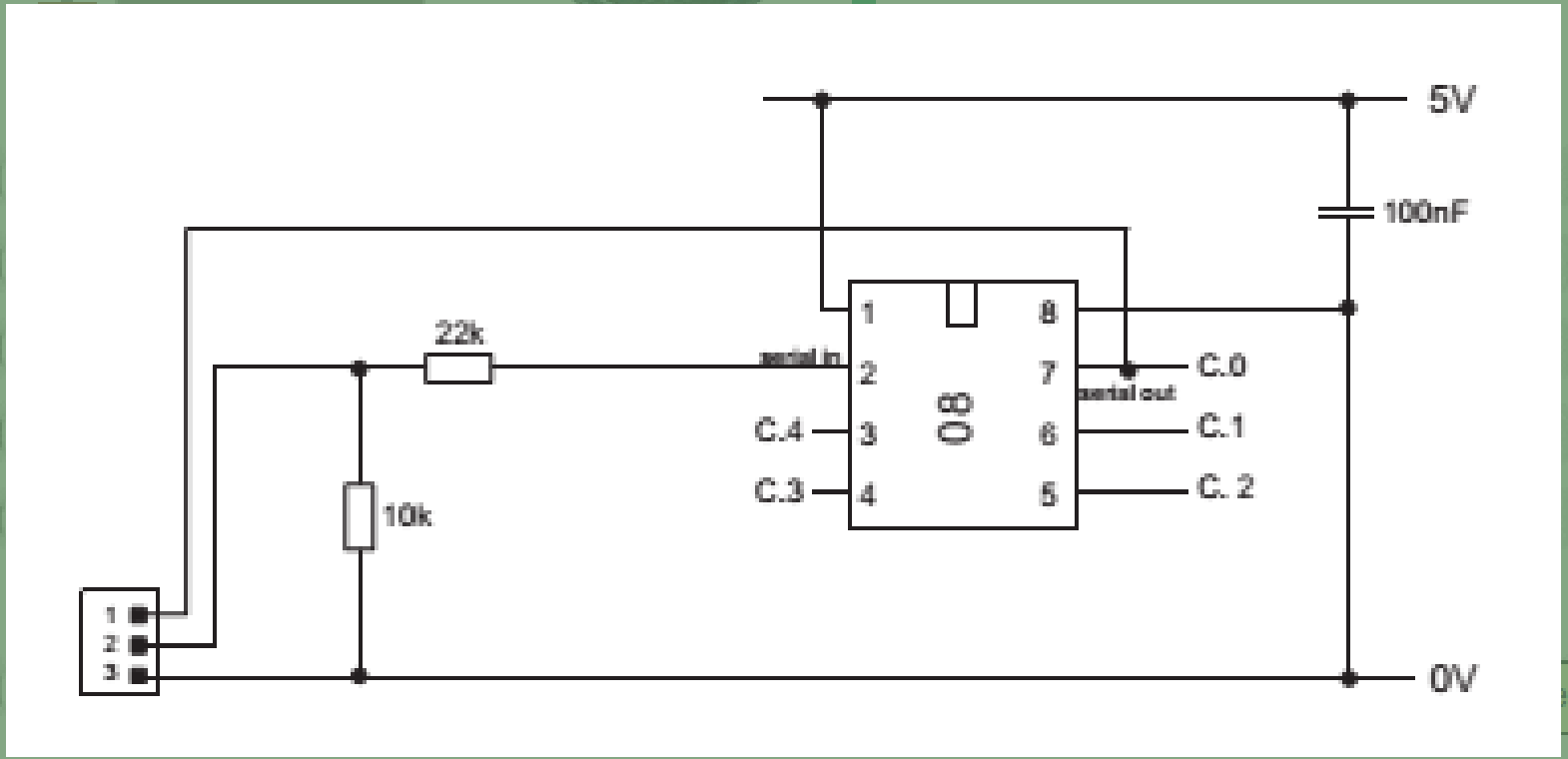
- These are not complete programs but “code tools” or modules of code you can reuse over and over rather than write new code. Simply change any variables to fit your program.

PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

MINIMUM HARDWARE REQUIRED FOR ANY PICAXE PROJECT:

PICAXE 08M2 (Other versions similar)



PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

MINIMUM HARDWARE REQUIRED FOR ANY PICAXE PROJECT:

- The 10K and 22K Resistors are required to insure reliable operation as the SER IN, which can double as in input pin, cannot be left floating, or else YOUR PROGRAM WILL NOT RUN.

PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

MINIMUM HARDWARE REQUIRED FOR ANY PICAXE PROJECT:

- You can use the SER OUT pin for an output port but it must be disconnected from the Serial programming port before using it as an output port. They cannot be connected simutaneously.
- The .1 uF capacitor is for stray oscillation and RF suppression. Good design practice.

PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

MINIMUM HARDWARE REQUIRED FOR ANY PICAXE PROJECT:

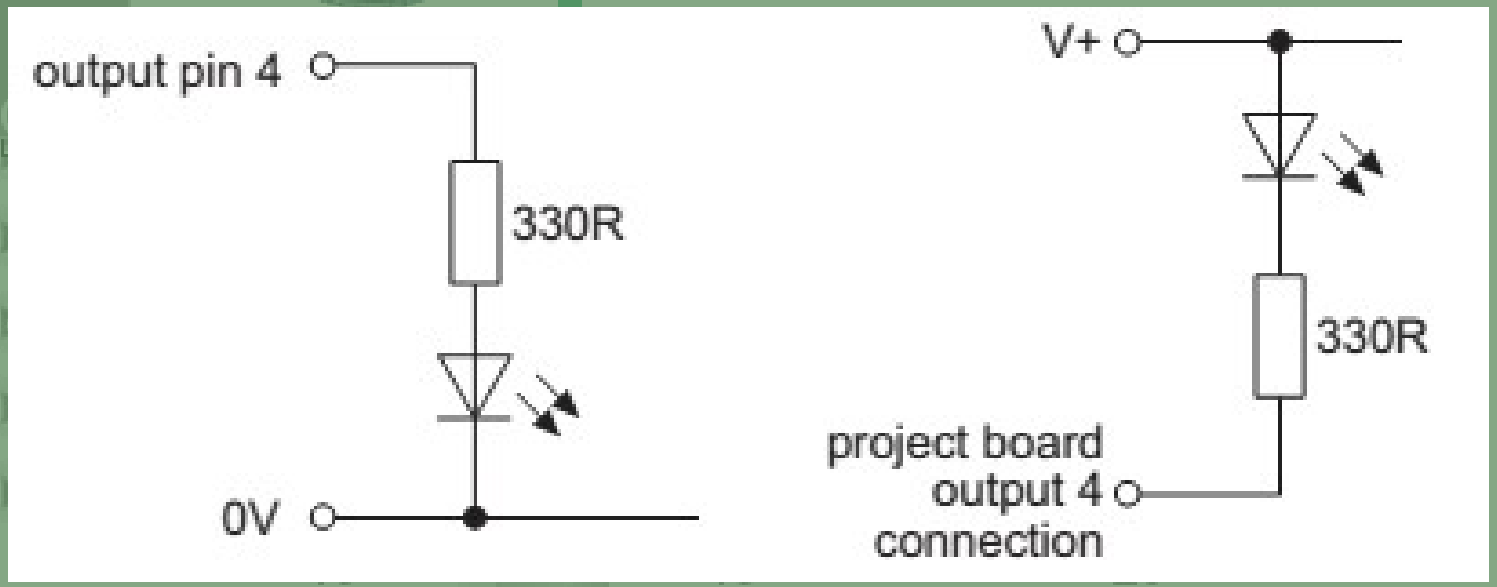
- The PICAXE does not require an external crystal for a clock. The PICAXE uses the internal 4-20 MHz clock (depending upon model) that it inherited from its PIC microcontroller base. Less parts count!!
- You can use an external resonator/crystal on the higher end PICAXE controllers.

PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

Programming Technique #1:
Turning LED, Lamp or Relay On and Off

```
main:
    high B.4
    pause 1000
    low B.4
    pause 1000
    goto main
```



PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

Programming Technique #2

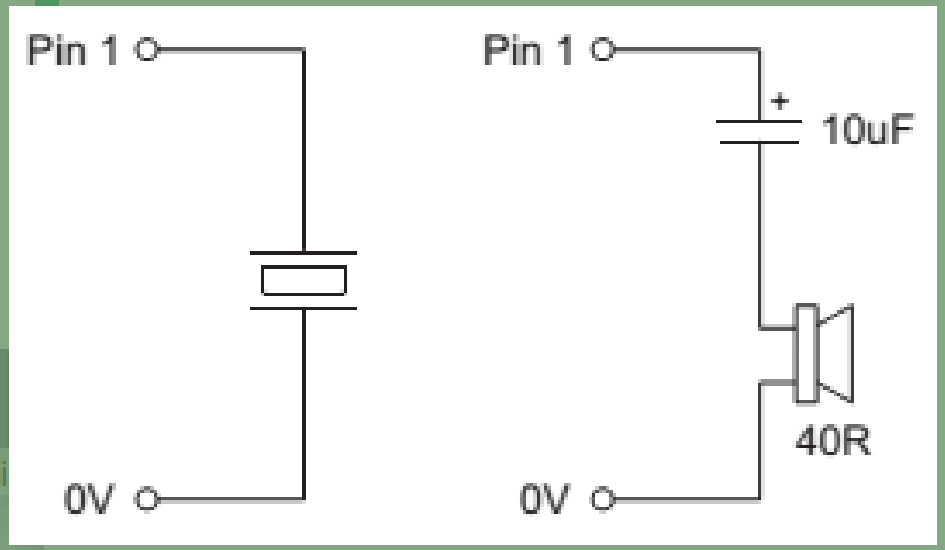
Generating Sound with a Piezo Buzzer

Sound 1, (100,50)

To produce a note of pitch 100, length 50 on pin 1

```
for b1 = 1 to 100
    sound 1, (b1,25)
next b1
```

To produce a varying noise using variable b1



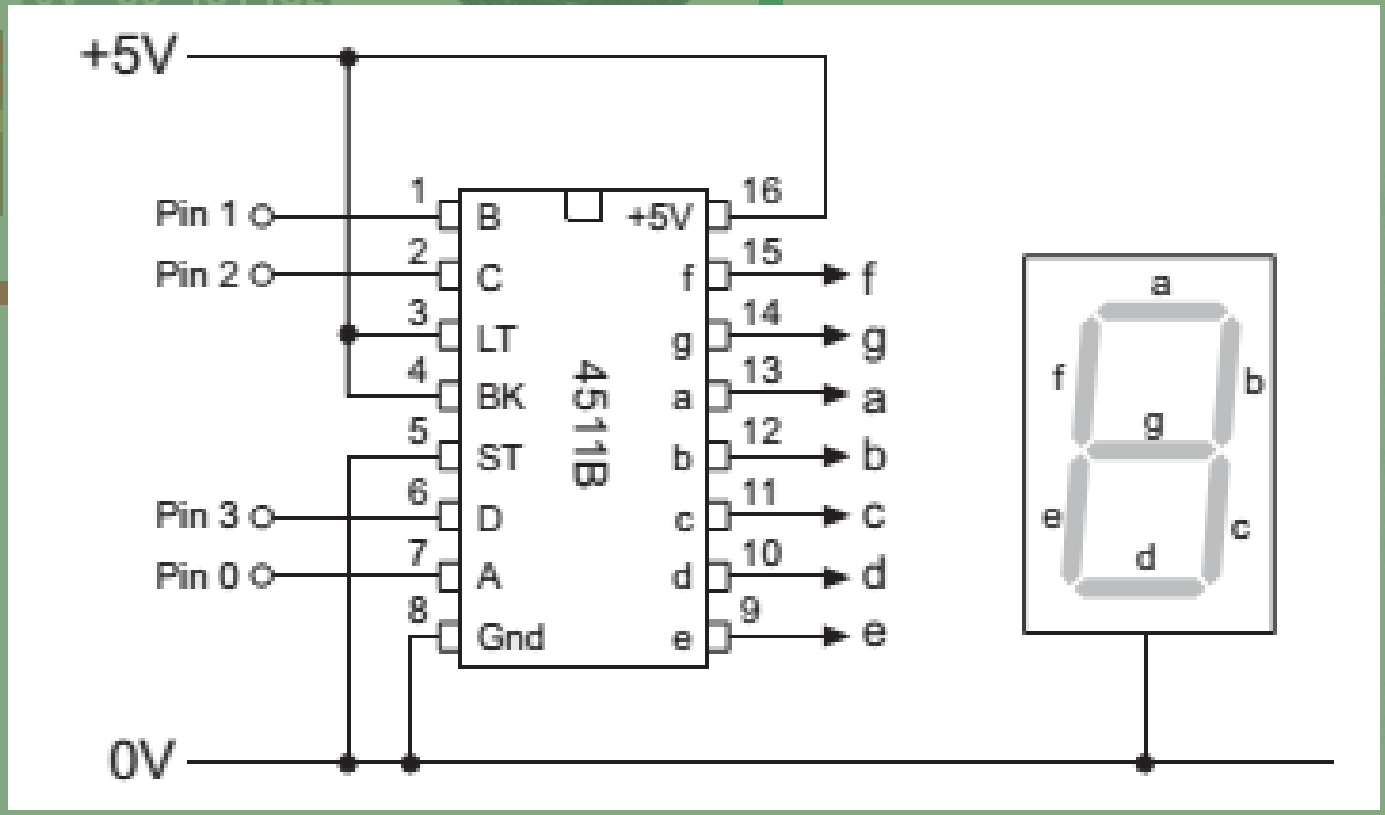
PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

Programming Technique #3

Seven Segment LED Interface

Sample Hardware Layout



PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

Programming Technique #3

Seven Segment LED Interface

Sample Code

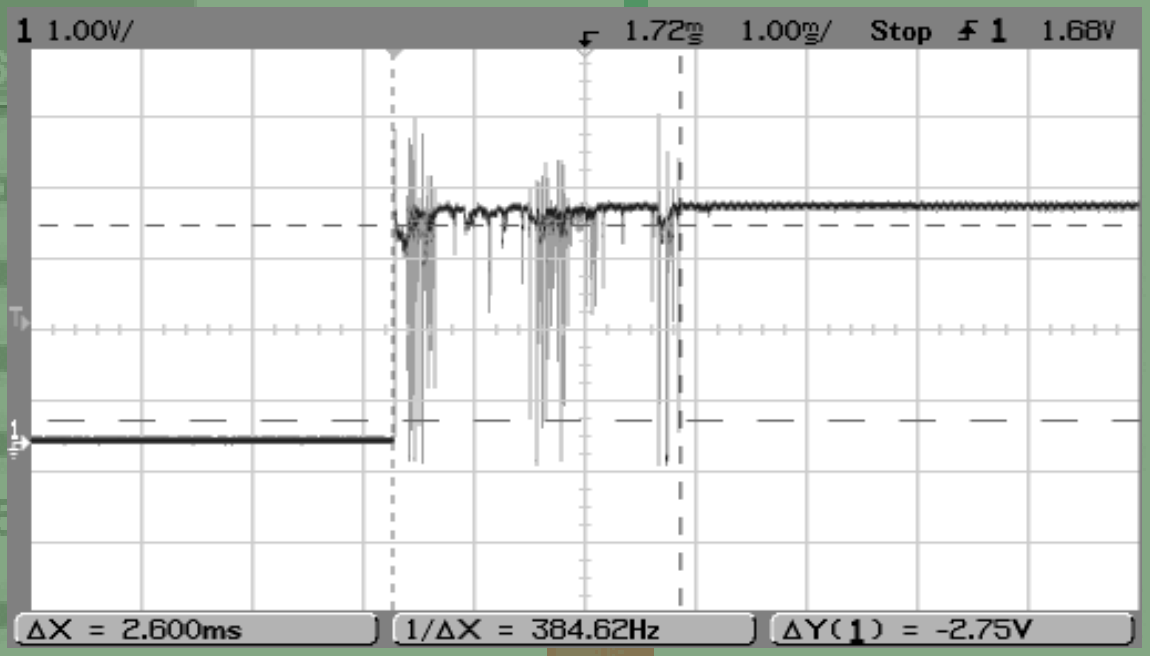
```
main:  for b1 = 0 to 9          ' Set up a for...next loop using variable b1
         let pins=b1            ' Output b1 onto the four data lines
        pause 1000            ' Pause 1 second
       next b1                 ' Next
       goto main               ' Loop back to start
```

PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

Programming Technique #4: Switch Debounce

What Happens when you press a switch?

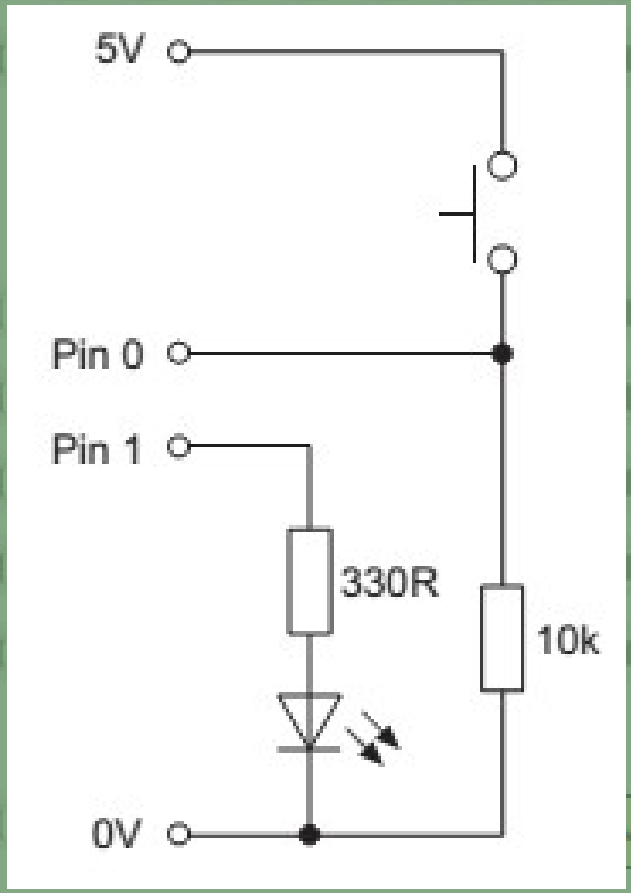


PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

Programming Technique #4: Switch Debounce

The next slide will illustate the effects of a switch before and after proper debouncing. The first group of code on the left is without debounce and the second group of code on the right is with debounce.



PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

Programming Technique #4: Switch Debounce

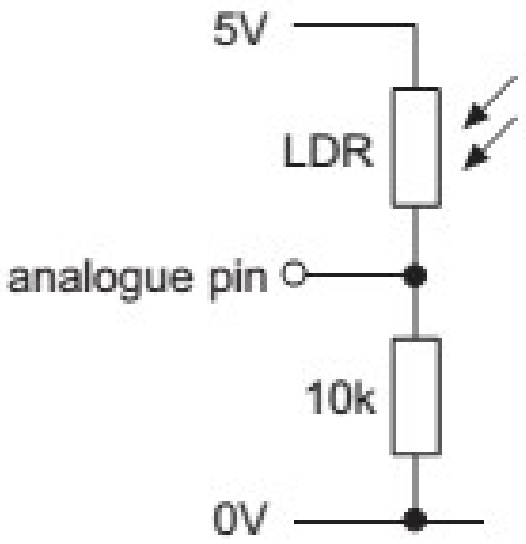
```
init:  let b0 = 0

main:  if pin 1 = 1 then add
       goto main

add:   let b0 = b0 + 1
       if b0 < 5 then main
       high 1
       goto main
```

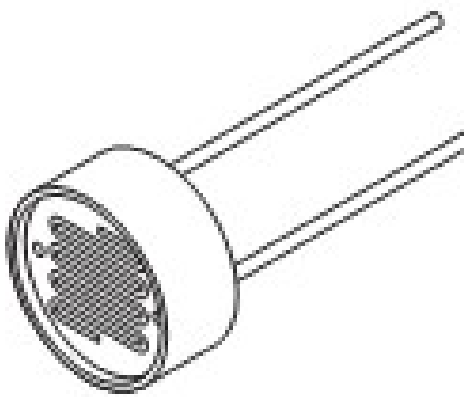
```
init:  let b0 = 0

main:  if pin 1 = 1 then add
       goto main

add:   pause 100     'short delay
       let b0 = b0 + 1
       if b0 < 5 then main
       high 1
       goto main
```

PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

Programming Technique #5: Reading a Sensor



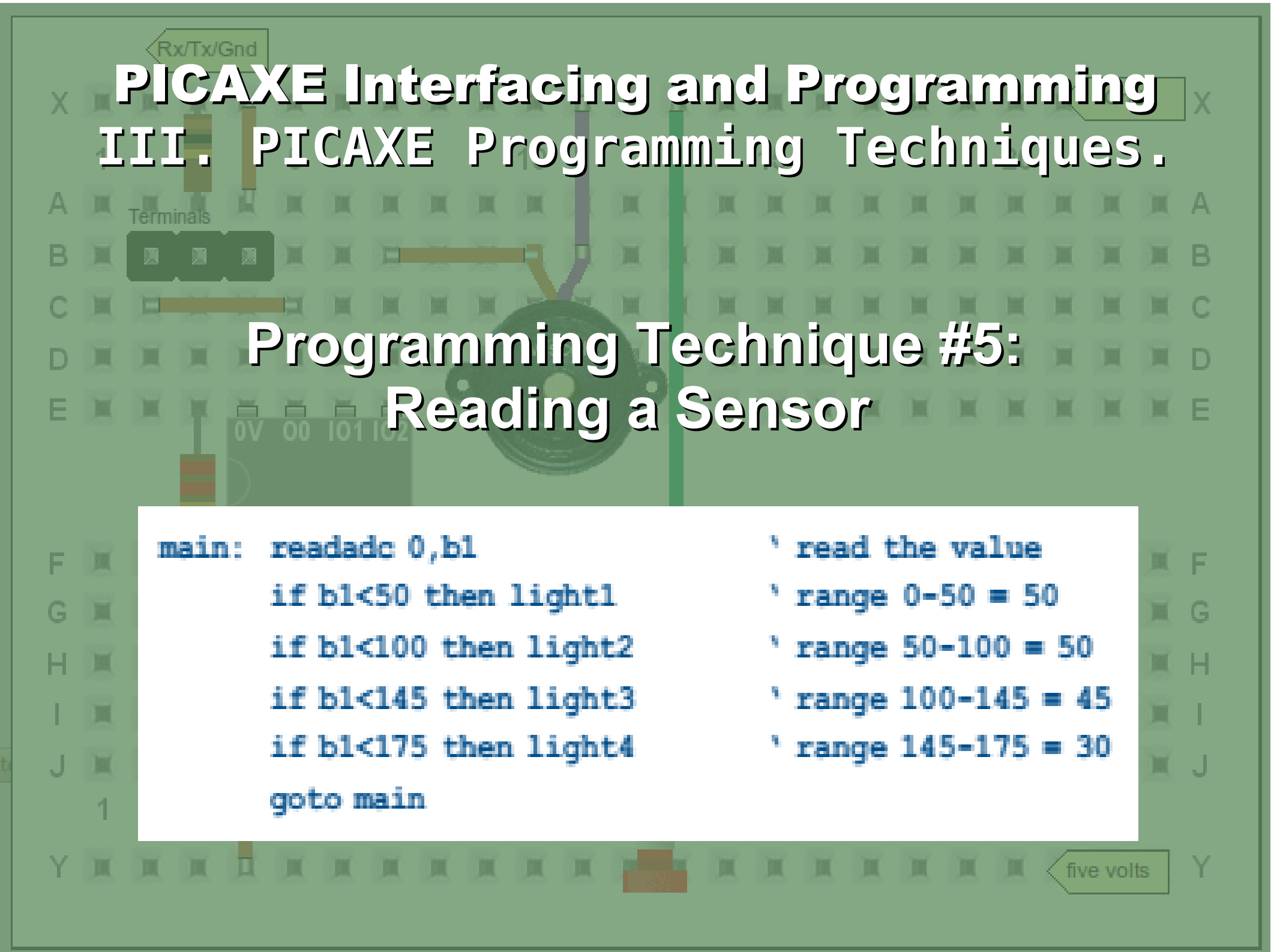


PICAXE Interfacing and Programming

III. PICAXE Programming Techniques.

Programming Technique #5: Reading a Sensor

```
main:  readadc 0,b1              ' read the value
       if b1<50 then light1      ' range 0-50 = 50
       if b1<100 then light2     ' range 50-100 = 50
       if b1<145 then light3     ' range 100-145 = 45
       if b1<175 then light4     ' range 145-175 = 30
       goto main
```



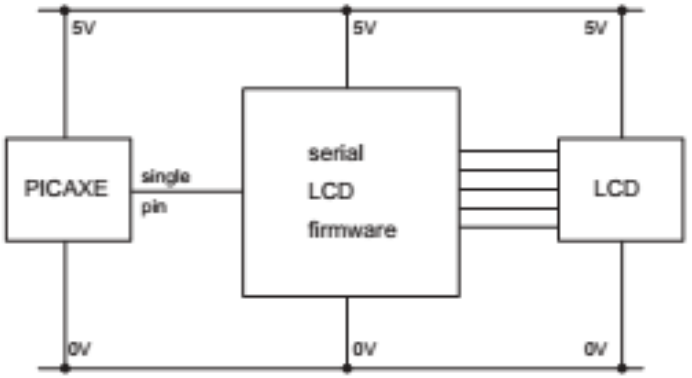
PICAXE Interfacing and Programming

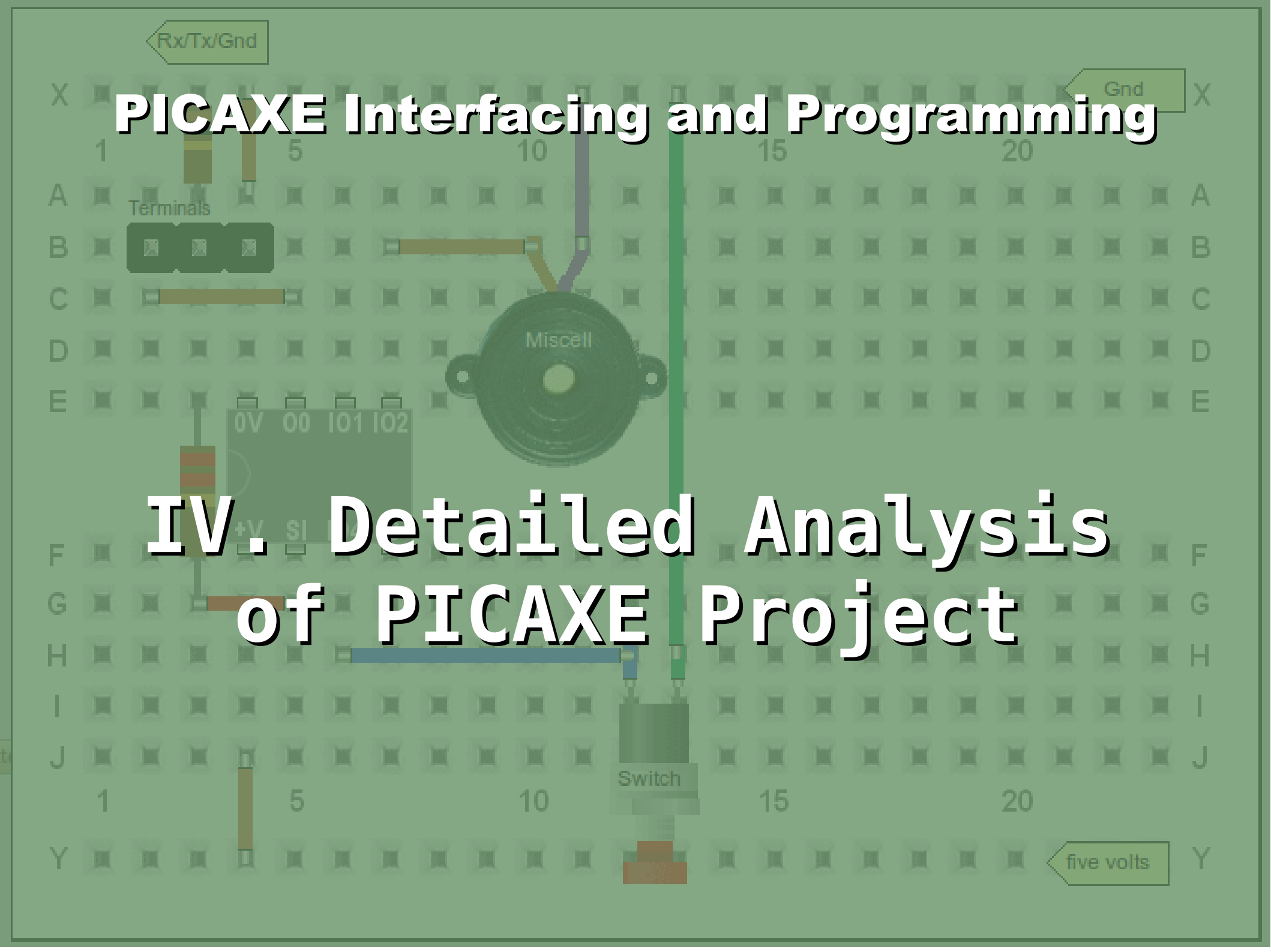
III. PICAXE Programming Techniques.

Programming Technique #6: Serial LCD (built in serial controller)

Main:

```
serout 1,N2400,(“Hello”) ‘ Send the word ‘Hello’
serout 1,N2400,(10,13) ‘ Send the ‘new line’ instructions.
pause 1000 ‘ Wait one second
goto main ‘ Loop back to the start
```

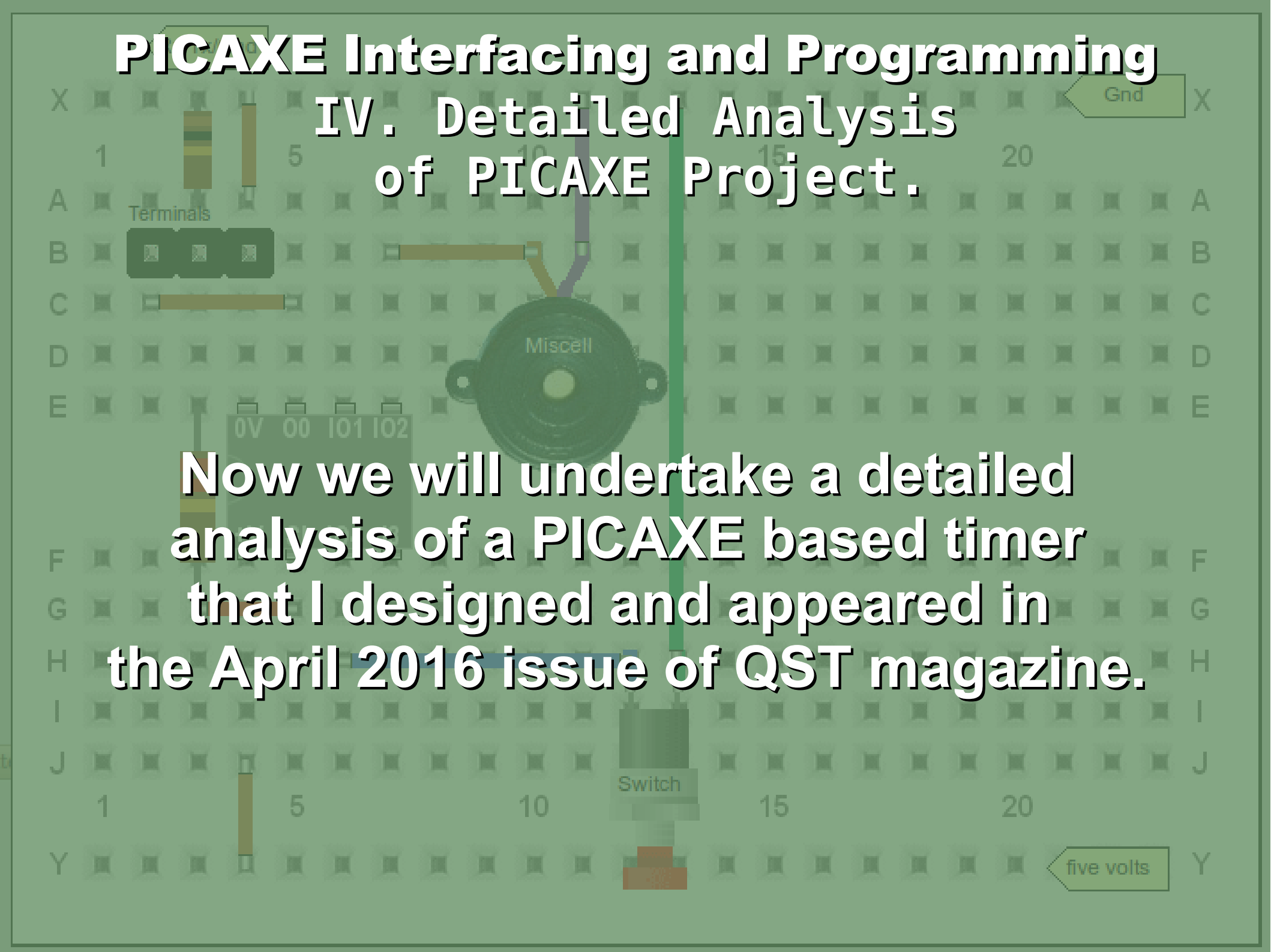




PICAXE Interfacing and Programming

IV. Detailed Analysis of PICAXE Project

PICAXE Interfacing and Programming
IV. Detailed Analysis of PICAXE Project.



Now we will undertake a detailed analysis of a PICAXE based timer that I designed and appeared in the April 2016 issue of QST magazine.

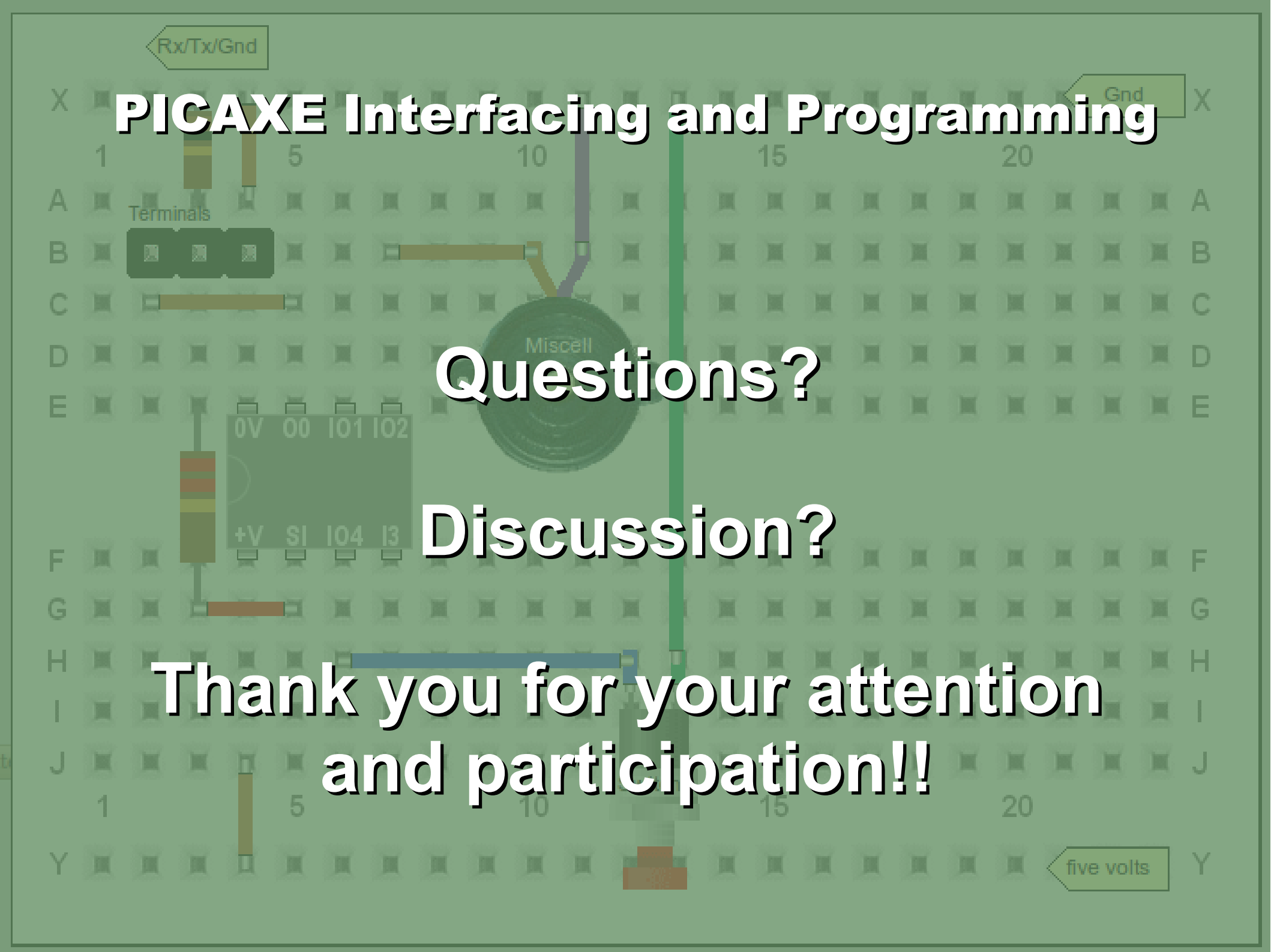
PICAXE Interfacing and Programming
IV. Detailed Analysis of PICAXE Project.

RESOURCES

Web: http://www.picaxe.com
Included PICAXE Manuals

Distributors in the U.S.:

Sparkfun: http://www.sparkfun.com/categories/124
Robotshop USA: http://www.robotshop.com/
PHAnderson http://www.phanderson.com/picaxe/
RobotMesh: http://www.robotmesh.com/picaxe



PICAXE Interfacing and Programming

Questions?

Discussion?

Thank you for your attention and participation!!

PICAXE Interfacing and Programming

CONTACT INFORMATION

Darrell Davis KT4WX
6350 Mills Road
Fort Meade, FL 33841-9584
(863) 245-9923

Email: kt4wx@arrl.net
Website: http://www.kt4wx.org